Withdrawn Draft

Warning Notice

The attached draft document has been withdrawn, and is provided solely for historical purposes. It has been superseded by the document identified below.

Withdrawal Date June 7, 2022

Original Release Date January 11, 2022

Superseding Document

Status	Final Public Draft (fpd)
Series/Number	NIST SP 800-160v1r1 fpd
Title	Engineering Trustworthy Secure Systems
Publication Date	June 7, 2022
DOI	https://doi.org/10.6028/NIST.SP.800-160v1r1.fpd
CSRC URL	https://csrc.nist.gov/publications/detail/sp/800-160/vol-1-rev-1/draft
Additional Information	

National Institute of Standards and Technology
U.S. Department of Commerce

Draft NIST Special Publication 800-160, Volume 1
Revision 1

Engineering Trustworthy Secure Systems

RON ROSS
MICHAEL McEVILLEY
MARK WINSTEAD

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-160v1r1-draft

National Institute of Standards and Technology
U.S. Department of Commerce

Draft NIST Special Publication 800-160, Volume 1
Revision 1

Engineering Trustworthy Secure Systems

RON ROSS
Computer Security Division
Information Technology Laboratory

MICHAEL McEVILLEY
MARK WINSTEAD
The MITRE Corporation
McLean, VA

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-160v1r1-draft

January 2022

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

AUTHORITY

This publication has been developed by NIST to further its statutory responsibilities under the Federal Information Security Modernization Act (FISMA), 44 U.S.C. § 3551 *et seq.*, Public Law (P.L.) 113-283. NIST is responsible for developing information security standards and guidelines, including minimum requirements for federal information systems, but such standards and guidelines shall not apply to national security systems without the express approval of the appropriate federal officials exercising policy authority over such systems. This guideline is consistent with requirements of the Office of Management and Budget (OMB) Circular A-130.

Nothing in this publication should be taken to contradict the standards and guidelines made mandatory and binding on federal agencies by the Secretary of Commerce under statutory authority. Nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, OMB Director, or any other federal official. This publication may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright in the United States. Attribution would, however, be appreciated by NIST.

National Institute of Standards and Technology Special Publication 800-160, Vol. 1, Rev. 1
Natl. Inst. Stand. Technol. Spec. Publ. 800-160, Vol. 1, Rev. 1, **207 pages** (January 2022)
CODEN: NSPUE2

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-160v1r1-draft

Certain commercial entities, equipment, or materials may be identified in this document to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

There may be references in this publication to other publications currently under development by NIST in accordance with its assigned statutory responsibilities. The information in this publication, including concepts, practices, and methodologies, may be used by federal agencies even before the completion of such companion publications. Thus, until each publication is completed, current requirements, guidelines, and procedures, where they exist, remain operative. For planning and transition purposes, federal agencies may wish to closely follow the development of these new publications by NIST.

Organizations are encouraged to review draft publications during the public comment periods and provide feedback to NIST. Many NIST publications, other than the ones noted above, are available at https://csrc.nist.gov/publications.

Public comment period: January 11, 2022 – February 25, 2022

Submit comments on this publication to: security-engineering@nist.gov

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive (Mail Stop 8930) Gaithersburg, MD 20899-8930

All comments are subject to release under the Freedom of Information Act (FOIA) [FOIA96].

REPORTS ON COMPUTER SYSTEMS TECHNOLOGY

The National Institute of Standards and Technology (NIST) Information Technology Laboratory (ITL) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology (IT). ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security of other than national security-related information in federal information systems. The Special Publication 800-series reports on ITL's research, guidelines, and outreach efforts in information systems security and privacy and its collaborative activities with industry, government, and academic organizations.

ABSTRACT

With the continuing frequency, intensity, and adverse consequences of cyber-attacks, disruptions, hazards, and other threats to federal, state, and local governments, as well as private sector organizations, the need for trustworthy secure systems has never been more important to the long-term economic and national security interests of the United States. Engineering-based solutions are essential to managing the complexity, dynamicity, and interconnectedness of today's systems, as exemplified by cyber-physical systems and systems-of-systems. This publication addresses the engineering-driven perspective and actions necessary to develop more defensible and survivable systems, inclusive of the machine, physical, and human components that compose those systems and the capabilities and services delivered by those systems. This publication starts with and builds upon established international standards for systems and software engineering by the International Organization for Standardization (ISO), the International Electrotechnical Commission (IEC), and the Institute of Electrical and Electronics Engineers (IEEE) and infuses systems security engineering methods, practices, and techniques into those systems and software engineering activities. The objective is to address security issues from a stakeholder protection needs, concerns, and requirements perspective and to use established engineering processes to help ensure that such needs, concerns, and requirements are addressed with appropriate fidelity and rigor throughout the system life cycle.

KEYWORDS

Assurance; developmental engineering; disposal; engineering trades; field engineering; implementation; information security; information security policy; inspection; integration; penetration testing; protection needs; requirements analysis; resilience; review; risk assessment; risk management; risk treatment; security architecture; security authorization; security design; security requirements; specifications; stakeholder; system of systems; system component; system element; system life cycle; systems; systems engineering; systems security engineering; trustworthiness; validation; verification.

ACKNOWLEDGMENTS

The authors gratefully acknowledge and appreciate the significant contributions from individuals and organizations in the public and private sectors whose constructive comments improved the overall quality, thoroughness, and usefulness of this publication. In particular, we wish to thank Jeff Brewer, Ken Cureton, Jordan Denmark, Rick Dove, Holly Dunlap, Jim Foti, Michael Hankins, Daryl Hild, M. Lee, Tom Llanso, Jimmie McEver, Perri Nejib, Cory Ocker, Daniel Patrick Pereira, Victoria Pillitteri, Greg Ritter, Thom Schoeffling, Theresa Soloway, Gary Stoneburner, Gregory Touhill, Isabel Van Wyk, Adam Williams, Drew Wilson, Carol Woody, William Young, and Michael Zisa. Finally, the authors wish to thank the students participating in the various INCOSE tutorials and MITRE Systems Security Engineering courses whose comments and insights help guide and inform many of the proposed changes in this publication.

HISTORICAL CONTRIBUTIONS

The authors gratefully acknowledge the contributions of Janet Carrier Oren, one of the original coauthors of NIST Special Publication 800-160, Volume 1. The authors also wish to acknowledge the following organizations and individuals for their historic contributions to this publication:

Organizations: National Security Agency; Naval Postgraduate School; Department of Defense Office of Acquisition, Technology, and Logistics; Department of Homeland Security Science and Technology Office, Cyber Security Division; International Council on Systems Engineering, United States Air Force; Air Force Institute of Technology; Northrop Grumman Corporation; The MITRE Corporation; Lockheed Martin Corporation.

Individuals: Beth Abramowitz, Max Allway, Kristen Baldwin, Dawn Beyer, Debora Bodeau, Paul Clark, Keesha Crosby, Judith Dahmann, Kelley Dempsey, Holly Dunlap, Jennifer Fabius, Daniel Faigin, Jeanne Firey, Robin Gandhi, Rich Graubart, Kevin Greene, Richard Hale, Daryl Hild, Kesha Hill, Danny Holtzman, Cynthia Irvine, Brett Johnson, Ken Kepchar, Stephen Khou, Alvi Lim, Logan Mailloux, Dennis Mangsen, Doug Maughn, Rosalie McQuaid, Joseph Merkling, John Miller, Thuy Nguyen, Perri Nejib, Lisa Nordman, Dorian Pappas, Paul Popick, Roger Schell, Thom Schoeffling, Matthew Scholl, Peter Sell, Gary Stoneburner, Glenda Turner, Edward Yakabovicz, and William Young.

Finally, the authors respectfully acknowledge the seminal work in computer security that dates back to the 1960s. The vision, insights, and dedicated efforts of those early pioneers in computer security serve as the philosophical and technical foundation for the security principles, concepts, methods, and practices employed in this publication to address the critically important problem of engineering trustworthy secure systems.

NOTES TO REVIEWERS

This update to SP 800-160, Volume 1 provided an excellent opportunity to reflect on the past five years of the publication's use by systems engineers and systems security engineers and to apply targeted lessons learned during that timeframe. In particular, we focused on the following strategic objectives which drove the majority of changes to the publication. These included:

- More strongly positioning Systems Security Engineering (SSE) as a sub-discipline of Systems Engineering (SE)
- Emphasizing that the responsibility for engineering trustworthy secure systems is not limited to security specialties and that the achievement of security outcomes must properly align with SE outcomes
- Aligning SSE practices with safety practices and other disciplines that deal with the loss of assets and the consequences of asset loss
- Focusing on the assurance of the correctness and effectiveness of the system's security capability to achieve authorized and intended behaviors and outcomes and control adverse effects and loss
- Emphasizing security roles and purpose to avoid inferring that SSE has responsibility for all aspects of security outcomes and prescribing what the SSE role is or should be
- More closely aligning to international standards

Based on the strategic objectives above, the significant revisions and enhancements to NIST's systems security engineering guidance include:

- A revised systems engineering and systems security engineering fundamentals section (Chapter Two) with new guidance on organizational assets and asset loss
- Simplified and streamlined system life cycle processes, structure, and associated security considerations (Chapter Three)
- A revised section on security policy and requirements (new Appendix C)
- A revised section on trustworthy secure design concepts for systems and system elements (new Appendix D)
- Enhanced security design principles presented in two distinct categories of trustworthiness and loss control (new Appendix E)
- A revised section on trustworthiness and assurance (new Appendix F)
- Selected modifications to the system life cycle processes (Chapter Three) to ensure consistency with ISO/IEC/IEEE 15288:202x
- Transitioning the content from two appendices, Summary of Systems Security Activities and Tasks (formerly Appendix D) and Roles, Responsibilities, and Skills (formerly Appendix E) to the NIST Systems Security Engineering web site

NIST is interested in your feedback on the specific changes made to the publication during this update. This can include the organization and structure of the publication, the presentation of the material, its ease of use, and the applicability of the technical content to current or planned systems engineering initiatives.

Thank you for taking the time to review the draft publication. Your comments can be sent to security-engineering@nist.gov using the comment template provided on the publication landing page at https://doi.org/10.6028/NIST.SP.800-160v1r1-draft.

Ron Ross
Project Leader,
Systems Security Engineering Project

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory (ITL) draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication or by reference to another publication. This call includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

    i) under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or

    ii) without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to: security-engineering@nist.gov.

TABLE OF CONTENTS

DISCLAIMER

This publication is intended to be used in conjunction with and as a supplement to **International Standard ISO/IEC/IEEE 15288**, *Systems and software engineering — System life cycle processes*. It is strongly recommended that organizations using this publication obtain the standard in order to fully understand the context of the security-related activities and tasks in each of the system life cycle processes. Content from the international standard that is referenced in this publication is used with permission from the Institute of Electrical and Electronics Engineers and is noted as follows: ***Reprinted with permission from IEEE, Copyright IEEE 2015, All rights reserved.***

The reprinted material has been updated to reflect any changes in the international standard.

ERRATA

This table contains changes that have been incorporated into Special Publication 800-160, Volume 1, Revision 1. Errata updates can include corrections, clarifications, or other minor changes in the publication that are either *editorial* or *substantive* in nature.

DATE	TYPE	REVISION	PAGE

PROLOGUE

"Providing satisfactory security controls in a computer system is in itself a system design problem. A combination of hardware, software, communications, physical, personnel and administrative-procedural safeguards is required for comprehensive security. In particular, software safeguards alone are not sufficient."

"Security Controls for Computer Systems," (The Ware Report), Rand Corporation
Defense Science Board Task Force on Computer Security, February 1970

"Mission assurance requires systems that behave with predictability and proportionality."

General Michael Hayden
Former NSA and CIA Director, Syracuse University, October 2009

"In the past, it has been assumed that to show that a system is safe, it is sufficient to provide assurance that the process for identifying the hazards has been as comprehensive as possible, and that each identified hazard has one or more associated controls. While historically this approach has been used reasonably effectively to ensure that known risks are controlled, it has become increasingly apparent that evolution to a more holistic approach is needed as systems become more complex and the cost of designing, building, and operating them become more of an issue."

Preface, NASA System Safety Handbook, Volume 1, November 2011

"This whole economic boom in cybersecurity seems largely to be a consequence of poor engineering."

Carl Landwehr
Communications of the ACM, February 2015

"Cybersecurity requires more than government action. Protecting our Nation from malicious cyber actors requires the Federal Government to partner with the private sector. The private sector must adapt to the continuously changing threat environment, ensure its products are built and operate securely, and partner with the Federal Government to foster a more secure cyberspace."

"Incremental improvements will not give us the security we need; instead, the Federal Government needs to make bold changes and significant investments in order to defend the vital institutions that underpin the American way of life."

Executive Order (EO) on Improving the Nation's Cybersecurity, May 2021

"[Systems] security engineering must be fundamental to systems engineering, not just a specialty discipline. Security concepts must be fundamental to [an] engineering education, and security proficiency must be fundamental in development teams. Security fundamentals must be clearly understood by stakeholders and effectively evaluated in a way that considers broad goals with security functions and outcomes."

Security in the Future of Systems Engineering [FUSE21]

FOREWORD

On May 12, 2021, the President signed an *Executive Order (EO) on Improving the Nation's Cybersecurity* [EO 14028]. The Executive Order stated—

> *"The United States faces persistent and increasingly sophisticated malicious cyber campaigns that threaten the public sector, the private sector, and ultimately the American people's security and privacy. The Federal Government must improve its efforts to identify, deter, protect against, detect, and respond to these actions and actors."*

The Executive Order further described the holistic nature of the cybersecurity challenges confronting the Nation with computing technology embedded in every type of system from general-purpose computing systems supporting businesses to cyber-physical systems controlling the operations in power plants that provide electricity to the American people. The Federal Government must bring to bear the full scope of its authorities and resources to protect and secure its computer systems, whether they are cloud-based, on-premises, or hybrid. The scope of protection and security must include systems that process data (information technology [IT]) and those that run the vital machinery that ensures our safety (operational technology [OT]).

To achieve this overarching objective, we must:

- Identify stakeholder assets and protection needs and provide protection commensurate with the criticality of those assets, needs, and the consequences of asset loss.
- Understand the modern threat space (i.e., adversary capabilities and intentions revealed by the targeting actions of those adversaries).
- Increase understanding of the growing complexity of systems to effectively reason about, manage, and address the uncertainty associated with that complexity.
- Adopt an engineering-based approach that addresses the principles of trustworthy secure design and apply those principles throughout the system life cycle.

Building trustworthy, secure systems cannot occur in a vacuum with isolated stovepipes for cyberspace, software, and information technology. Rather, it requires a holistic approach to protection, broad-based thinking across all assets where loss could occur, and an understanding of adversity, including how adversaries attack and compromise systems. As such, this publication addresses considerations for the engineering-driven actions necessary to develop defensible and survivable systems, including the components that compose and the services that depend on those systems. The publication builds upon a set of international standards for systems and software engineering published by the International Organization for Standardization (ISO), the International Electrotechnical Commission (IEC), and the Institute of Electrical and Electronics Engineers (IEEE) and infuses systems security engineering techniques, methods, and practices into those systems and software engineering activities. The overall objective is to address security issues from a stakeholder requirements and protection needs perspective and to use established engineering processes to ensure that such requirements and needs are addressed with appropriate fidelity and rigor across the entire life cycle of the system.

Engineering trustworthy, secure systems is a significant undertaking that requires a substantial investment in the requirements, architecture, and design of systems, components, applications, and networks. A trustworthy system is a system that provides compelling evidence to support claims that it meets its requirements to deliver the protection and performance needed by stakeholders within their defined tolerance of risk. Introducing a disciplined, structured, and standards-based set of systems security engineering activities and tasks provides an important starting point and forcing function to initiate needed change.

> “Some have a tendency to dismiss ideas that are older than x years, where x seems to be getting smaller and smaller as the pace of technology development continues to increase at an exponential rate. There is a tendency among some to think that cybersecurity is purely a technology problem – that if you just build the right widgets (device or piece of software), the problem will be solved. I call this the 'widget mentality.' Widgets are certainly important, but knowledge and deep understanding are essential. Indeed, developing widgets without an understanding of the nature of the problem and what constitutes a real solution to the problem is ineffective. *[It is important to understand*] …the form of principles that underlie cybersecurity so that designers can understand what widgets to build, to what requirements they should build them, how they should be deployed and interconnected within cyberspace, and how to operate them when under attack.”
>
> -- **O. Sami Saydjari**
> ***Engineering Trustworthy Systems*** **[Saydjari18]**

THE POWER OF SCIENCE AND ENGINEERING

When crossing a bridge, we have a reasonable *expectation* that the bridge will not collapse and will get us to our destination without incident. For bridge builders, the focus is on equilibrium, static and dynamic loads, vibrations, and resonance. The science of *physics* combines with civil engineering principles and concepts to produce a product that we deem *trustworthy*, giving us a level of confidence that the bridge is fit-for-purpose.

For system developers, there are also fundamental principles and concepts that can be found in *mathematics*, *computational science*, *computer and electrical engineering*, *systems engineering,* and *software engineering* that when properly employed, provide the necessary and sufficient trustworthiness to give us that same level of confidence. Trustworthy secure systems cannot be achieved simply by applying best practices in cyber hygiene. Rather, it will take a significant and substantial investment in strengthening the underlying systems and system components by employing transdisciplinary systems engineering efforts guided and informed by well-defined security requirements and secure architectures and designs. Such efforts have been proven over time to produce sound engineering-based solutions to complex and challenging systems security problems. Only under those circumstances can we build systems that are adequately secure and exhibit a level of trustworthiness that is sufficient for the purpose for which the system was built.

HOW TO USE THIS PUBLICATION

This publication is intended to serve as a *reference* and *educational resource* for engineers and engineering specialties, architects, designers, and individuals involved in the development of trustworthy secure systems and system components. There is no expectation that all of the security considerations, system life cycle processes, design principles, or other technical content in this publication will be employed in systems engineering processes. Rather, the material can be applied selectively by organizations, individuals, or engineering teams to improve the security and trustworthiness of systems and system components.

CHAPTER ONE

INTRODUCTION

THE NEED FOR SYSTEMS ENGINEERING-BASED TRUSTWORTHY SECURE SYSTEMS

The need for trustworthy secure systems[1] stems from the adverse effects associated with a diverse set of stakeholder needs that are driven by mission, business, and other objectives and concerns. The characteristics of these systems reflect a growth in the geographic size, number, and types of components and technologies[2] that compose the systems; the complexity and dynamicity in the behaviors and outcomes of the systems; and the increased dependence that results in a range of consequences from major inconvenience to catastrophic loss due to adversity[3] within the global operating environment. Today's systems have the dimensions and inherent complexity that require a disciplined and structured engineering approach to achieve any expectation that the complexity can be effectively managed and that the systems can be demonstrated to be trustworthy secure within the practical and feasible limits of human capability and certainty.

Managing the complexity of systems and being able to claim that those systems are trustworthy secure means that, first and foremost, there must be a level of confidence in the feasibility, correctness-in-concept, philosophy, and design regarding the ability of a system to produce only the intended behavior and outcomes. That basis provides the foundation to address security concerns with sufficient confidence that the system functions only as intended while subjected to a spectrum of adversity and to realistically bound those expectations with respect to constraints and uncertainty. The failure to address this complexity will continue to leave the Nation susceptible to the consequences of adversity with the potential for causing serious, severe, or even catastrophic consequences.

Security is freedom from the conditions that can cause a loss of *assets* with unacceptable consequences.[4] The scope of security must be defined by stakeholders in terms of the assets to which security applies and the consequences against which security is assessed.[5]

---

[1] A *system* is an arrangement of parts or elements that exhibit a behavior or meaning that the individual constituents do not [INCOSE19]. The elements that compose a system include hardware, software, data, humans, processes, procedures, facilities, materials, and naturally occurring entities [ISO 15288]. Examples of systems include financial systems, manufacturing systems, transportation distribution systems, logistics systems, vehicular systems, mobile devices, Internet of Things (IoT) devices, weapons systems, space systems, environmental control systems, communications systems, cyber-physical systems, and industrial control systems.

[2] The term *technology* is used in the broadest context in this publication to include computing, communications, and information technologies, as well as any mechanical, hydraulic, pneumatic, or structural components in systems that contain or are enabled by such technologies. This view of technology provides an increased recognition of the digital, computational, and electronic machine-based foundation of modern complex systems and the growing importance of the trustworthiness of that foundation in providing the system's functional capability and explicit interaction with its physical machine and human system elements.

[3] The term *adversity* refers to those conditions that can cause a loss of assets (e.g., threats, attacks, vulnerabilities, hazards, disruptions, and exposures).

[4] The phrasing used in this definition of *security* is intentional. [Anderson20] noted that "now that everything's acquiring connectivity, you can't have safety without security, and these ecosystems are emerging." Reflecting this observation, the security definition was chosen to achieve alignment with a prevailing *safety* definition.

[5] Adapted from [NASA11].

Systems engineering provides the foundation for a disciplined and structured approach to building trustworthy secure systems. Trustworthiness[6] is defined in [Neumann04] as follows:

> *By trustworthiness, we mean simply worthy of being trusted to fulfill whatever critical requirements may be needed for a particular component, subsystem, system, network, application, mission, enterprise, or other entity. Trustworthiness requirements might typically involve (for example) attributes of security, reliability, performance, and survivability under a wide range of potential adversities. Measures of trustworthiness are meaningful only to the extent that the requirements are sufficiently complete and well defined, and can be accurately evaluated.*

Systems security engineering is considered a subdiscipline of systems engineering. It provides considerations for the security-oriented activities and tasks that produce security outcomes as part of every systems engineering process activity with emphasis on the appropriate level of fidelity and rigor needed to achieve assurance and trustworthiness objectives. Systems security engineering provides the needed complementary engineering capability that extends the notion of trustworthiness to deliver trustworthy secure systems. Trustworthy secure systems are less susceptible but not impervious to the effects of modern adversities. Such adversities come in malicious and non-malicious forms and can emanate from a variety of sources including physical and electronic. Adversities can include attacks from determined and capable adversaries, human errors of omission or commission, accidents and incidents, component faults and failures, abuse and misuse, and natural or human-made disasters.

1.1 PURPOSE AND APPLICABILITY

The purpose of this publication is:

- To provide a basis to formalize a discipline for systems security engineering in terms of its principles, concepts, and activities
- To foster a common mindset to deliver security for any system, regardless of its purpose, type, scope, size, complexity, or stage of the system life cycle
- To provide considerations and to demonstrate how systems security engineering principles, concepts, and activities can be effectively applied to systems engineering activities
- To advance the field of systems security engineering as a discipline that can be applied and studied
- To serve as a basis for the development of educational and training programs, including the development of individual certifications and other professional assessment criteria

The considerations set forth in this publication are applicable to all federal systems other than those systems designated as national security systems as defined in 44 U.S.C., Section 3542.[7] These considerations have been broadly developed from a technical and technical management perspective to complement similar considerations for national security systems and may be

[6] *Trustworthiness* is not only about demonstrably meeting a set of requirements, but the requirements must also be complete, consistent, and correct. From a security perspective, a trustworthy system is a system that meets a set of well-defined requirements including security requirements.

[7] [OMB M-19-03] states that increasing the trustworthiness of systems is a significant undertaking that requires a substantial investment in the requirements, architecture, design, and development of systems, system components, applications, and networks. The policy requires federal agencies to implement the systems security engineering principles, concepts, techniques, and system life cycle processes in this publication for all high value assets (HVA).

used for such systems with the approval of federal officials exercising policy authority over such systems. State, local, and tribal governments, as well as private sector entities, are encouraged to consider using the material in this publication, as appropriate.

The applicability statement is not meant to limit the technical and management application of these considerations. That is, the security design principles, concepts, and techniques described in this publication are part of a *trustworthy secure design* approach as described in Appendix D and can be applied to any type of system, including:

- **New Systems**

  The engineering effort includes such activities as concept exploration, preliminary or applied research to refine the concepts and/or feasibility of technologies employed in a new system, and an assessment of alternative solutions. This effort is initiated during the concept and development stages of the system life cycle.

- **Dedicated or Special-Purpose Systems**

  - *Security-dedicated or security-purposed systems:* The engineering effort delivers a system that satisfies a security-dedicated need or provides a security-oriented purpose and does so as a stand-alone system that may monitor or interact with other systems. Such systems can include surveillance systems, physical protection systems, monitoring systems, and security service provisioning systems.

  - *High-confidence, dedicated-purpose systems:* The engineering effort delivers a system that satisfies the need for real-time control of vehicles, industrial or utility processes, or weapons, nuclear, and other special-purpose needs. Such systems may include multiple operational states or modes with varying forms of manual, semi-manual, automated, or autonomous modes. These systems have highly deterministic properties, strict timing constraints and functional interlocks, and severe or catastrophic consequences of failure.

- **System of Systems**

  The engineering effort occurs across a set of constituent systems, each system with its own stakeholders, primary purpose, and planned evolution. The composition of the constituent systems into a *system of systems* [Maier98] produces a capability that would otherwise be difficult or impractical to achieve. This effort can occur across a variety of system of systems from a relatively informal, unplanned system of systems concept and evolution that emerges over time via voluntary participation to a more formal execution with the most formal being a system of systems concept that is directed, structured, and planned, and achieved via a centrally managed engineering effort. Any resulting emergent behavior often introduces opportunities and additional challenges for systems security engineering.

The design principles, concepts, and techniques can also be applied at any stage in the system life cycle when an engineered approach is needed to achieve any of the following objectives:

- **System Modifications**

  - *Reactive modifications to fielded systems:* The engineering effort occurs in response to adversity that diminishes or prevents the system from achieving the design intent. This effort can occur during the production, utilization, or support stages of the system life cycle and may be performed concurrently with or independent of day-to-day system operations.

- *Planned upgrades to fielded systems while continuing to sustain day-to-day operations:* The planned system upgrades may enhance an existing system capability, provide a new capability, or constitute a technology refresh of an existing capability. This effort occurs during the production, utilization, or support stages of the system life cycle.
- *Planned upgrades to fielded systems that result in new systems:* The engineering effort is carried out as if developing a new system with a system life cycle that is distinct from the life cycle of a fielded system. The upgrades are performed in a development environment that is independent of the fielded system.

- **System Evolution**

  The engineering effort involves migrating or adapting a system or system implementation from one operational environment or set of operating conditions to another operational environment or set of operating conditions.[8]

- **System Retirement**

  The engineering effort removes system functions or services and system elements from operation, including removal of the entire system, and may also include the transition of system functions and services to another system. The effort occurs during the retirement stage of the system life cycle and may be carried out while sustaining day-to-day operations.

1.2 TARGET AUDIENCE

This publication is intended for security engineering and other engineering professionals who accomplish the activities and tasks that are defined by the system life cycle processes described in Chapter Three. The term *systems security engineer* is specifically used to include security professionals who perform the activities and tasks described in this publication. It may apply to an individual or a team of individuals from the same organization or different organizations.[9] This publication can also be used by professionals who perform other system life cycle activities or activities related to the education and/or training of systems engineers and systems security engineers. These include but are not limited to:

- Individuals with systems engineering, software engineering, architecture, design, development, and integration responsibilities
- Individuals with security governance, risk management, and oversight responsibilities
- Individuals with security verification, validation, testing, evaluation, auditing, assessment, inspection, and monitoring responsibilities
- Individuals with acquisition, budgeting, and project management responsibilities
- Individuals with system security administration, operations, maintenance, sustainment, logistics, and support responsibilities

[8] Increasingly, there is a need to reuse or leverage system implementation successes within operational environments that are different from how they were originally designed and developed. This type of reuse or reimplementation of systems within other operational environments is more efficient and represents potential advantages in maximizing interoperability between various system implementations.

[9] Systems security engineering activities and tasks can be applied to a mechanism, component, system element, system, system of systems, processes, or organizations. Regardless of the size or complexity of the entity, there is need for a transdisciplinary systems engineering team to deliver systems that are trustworthy and that satisfy the protection needs and concerns of stakeholders. The processes are intended to be tailored to facilitate effectiveness.

- Providers of technology products, systems, or services
- Academic institutions offering systems/computer/security engineering programs.

> *"Security is embedded in systems. Rather than two engineering groups designing two systems, one intended to protect the other, systems engineering specifies and designs a single system with security embedded in the system and its components."*
>
> -- **An Objective of Security in the Future of Systems Engineering** [FUSE21]

1.3 HOW TO USE THIS PUBLICATION

Organizations using this guidance for their systems security engineering efforts can select and employ some or all of the 30 [ISO 15288] processes and some or all of the security-related activities and tasks defined for each process. There are process dependencies, and the successful completion of some activities and tasks necessarily invokes other processes or leverages the results of other processes. This publication is intended to be flexible in its application in order to meet the diverse needs of organizations. It is *not* intended to provide a recipe or roadmap for execution. Rather, it can be viewed as a catalog for achieving the security outcomes of a systems engineering perspective on system life cycle processes – relying on the experience and expertise of the engineering organization to determine what is correct for its purpose.

The system life cycle processes can take advantage of any system or software development methodology, including *waterfall*, *spiral*, *DevOps*, or *agile*. In addition, the processes can be applied recursively, iteratively, concurrently, sequentially, or in parallel and to any system regardless of its size, complexity, purpose, scope, environment of operation, or special nature. The full extent of the application of the content in this publication is guided and informed by stakeholder capability needs, protection needs, and concerns with particular attention paid to considerations of cost, schedule, and performance.

1.4 ORGANIZATION OF THIS PUBLICATION

The remainder of this publication is organized as follows:

- **Chapter Two** provides an overview of the foundational concepts and principles of systems engineering and the specialty discipline of systems security engineering. It presents the basic concepts associated with a system; addresses the concepts of loss, security, protection needs and assets; explains how system security is demonstrated using the concepts of trustworthiness and assurance; and introduces a framework for implementing systems security engineering.
- **Chapter Three** describes security considerations, contributions, and extensions to the system life cycle processes defined in the international systems and software engineering standard [ISO 15288]. Each of the system life cycle processes contains a set of security enhancements that augment or extend the process outcomes, activities, and tasks defined by the standard. The enhanced processes address system security as they are applied throughout the system life cycle.

- The following sections provide additional information for the effective application of the activities and tasks in this publication:
  - References
  - Appendix A: Glossary
  - Appendix B: Acronyms
  - Appendix C: Security Policy and Requirements
  - Appendix D: Trustworthy Secure Design
  - Appendix E: Principles for Trustworthy Secure Design
  - Appendix F: Trustworthiness and Assurance

A SECURITY ENGINEERING FOCUS

This publication does not focus exclusively on cybersecurity but instead, addresses **security** more broadly. Given the scope of this publication, the following observations are relevant and worth noting:

"For the first few decades as a burgeoning discipline, cybersecurity has been dominated by the development of widgets to address some aspect of the problem. Systems have become increasingly complex and interconnected, creating even more attack opportunities, which in turn creates even more opportunities to create defensive widgets that will bring some value in detecting or preventing an aspect of the attack space. Eventually, this becomes a game of whack-a-mole in which a simulated mole pops up from one of many holes and the objective is to whack the mole before it pops back in its hole. The moles represent new attacks, and the holes represent a huge array of potential vulnerabilities—both known and as-yet-undiscovered."

"Underlying [the discipline of] engineering is science. Sometimes engineering gets ahead of science, such as in bridge building, where the fundamentals of material science were not well understood. Many bridges were built; many fell down; some stayed up; designs of the ones that stayed up were copied. Eventually, for engineering to advance beyond some point, science must catch up with engineering. The science underlying cybersecurity [and more generally, security] engineering is complex and difficult. On the other hand, there is no time like the present to start, because it is both urgent and important to the future…"

-- **O. Sami Saydjari**
***Engineering Trustworthy Systems* [Saydjari18]**

ENGINEERING-DRIVEN SOLUTIONS

The effectiveness of any engineering discipline first requires a thorough understanding of the problem and consideration of all feasible solutions before acting to solve the identified problem. To maximize the effectiveness of systems security engineering, the security requirements for the protection against asset loss must be driven by business, mission, and all other stakeholder asset loss concerns. The security requirements are defined and managed as a well-defined set of engineering requirements and cannot be addressed independently or after the fact.

In the context of systems security engineering, the term *protection* has a broad scope and is primarily focused on the concept of assets and asset loss. The protection capability provided by a system goes beyond prevention and aims to control the events, conditions, and consequences that constitute asset loss. It is achieved in the form of the specific capability and constraints on system architecture, design, function, implementation, construction, selection of technology, methods, and tools and must be "engineered in" as part of the system life cycle process.

Understanding stakeholder asset protection needs (including assets that they own and assets that they do not own but must protect) and expressing those needs through a set of well-defined security requirements is an investment in the organization's mission and business success in the modern age of global commerce, powerful computing systems, and network connectivity.

CHAPTER TWO

THE FUNDAMENTALS

THE CONCEPTS ASSOCIATED WITH SYSTEMS AND SECURITY ENGINEERING

This chapter provides the foundations of systems engineering and systems security engineering; presents the basic concepts associated with a system, including system structure, types of systems, and system of systems; offers a perspective on system security that addresses the concepts of loss, security, protection needs, and assets; describes how system security is demonstrated; and introduces a framework for implementing systems security engineering.

2.1 ENGINEERING FOUNDATIONS

Systems engineering is a transdisciplinary and integrative approach to enabling the successful realization, use, and retirement of engineered systems. It employs systems principles and concepts, as well as scientific, technological, and management methods to achieve such systems [INCOSE]. Systems engineering uses a collection of technical and non-technical system life cycle processes with associated activities and tasks. The technical processes apply engineering analysis and design principles to realize and deliver a system with the capability to satisfy stakeholder needs and associated emergent properties.[10] The non-technical processes provide engineering management of all aspects of the engineering project, agreements between parties involved in the project, and project-enabling support to facilitate execution of the project.

Systems engineering is *system-holistic* in nature, whereby the contributions across multiple engineering and specialty disciplines are evaluated and balanced to produce a coherent capability that is the *system*. Systems engineering applies both systems science and systems thinking[11] to solve problems and balances the often-conflicting needs, priorities, and constraints of performance, cost, schedule, and effectiveness to optimize the objectives for the solution with an acceptable level of uncertainty. Systems engineering is *outcome-oriented* and leverages a flexible set of engineering processes to realize a system while effectively managing complexity and serving as the principal integrating mechanism for the technical, management, and support activities related to the engineering effort. Finally, systems engineering is *data*- and *analytics-driven* to ensure that all decisions and trades are guided and informed by data produced by analyses conducted with an appropriate level of fidelity and rigor.

[10] An *emergent property* is a property occurring, or emerging, due to interactions among entities within the system and often outside of the system. Emergent properties are typically qualitative in nature, subjective in their nature and assessment, and require consensus agreement based on evidentiary analysis and reasoning. Emergent properties may be anticipated or unanticipated and may be beneficial or detrimental. Emergent properties of systems include safety, security, survivability, maintainability, resilience, reliability, agility, and availability. INCOSE identifies specialty engineering disciplines within systems engineering that are necessary to deliver a complete system, some of which address one or more system emergent properties.

[11] *Systems science* is an interdisciplinary field that studies complex systems in nature, society, and science. It aims to develop interdisciplinary foundations that are applicable in a variety of areas, such as social sciences, engineering, biology, and medicine. *Systems thinking* is a discipline of examining wholes, interrelationships, and patterns [SEBOK].

Systems engineering efforts are complex, requiring close coordination between the *engineering team* and stakeholders throughout the various stages of the system life cycle.[12] While systems engineering is typically considered in terms of its developmental role as part of the acquisition of a capability, systems engineering efforts and responsibilities do not end once a system completes development and is transitioned to the environment of operation for day-to-day operational use. Stakeholders responsible for the utilization, support, and retirement of a system provide data to the systems engineering team on an ongoing basis. This data captures their experiences, problems, and issues associated with the use and sustainment of the system. Stakeholders also advise on enhancements and improvements made or that they wish to see incorporated into system revisions. In addition, field engineering (also known as sustainment engineering) provides on-site, full life cycle engineering support for operations, maintenance, and sustainment organizations. Field engineering teams coexist with or are dispatched to operational sites and maintenance depots to provide continuous systems engineering support.

ENGINEERING THE RIGHT SOLUTIONS FOR THE RIGHT REASONS

NASCAR is an organization that governs competition among race teams that engineer, operate, and sustain high-performance racecars designed to be extremely fast, able to operate in hostile racing environments, and able to protect the teams' most critical asset – the driver. These racecars are very different from the typical family car that carries your kids to school or makes the trip to the grocery store. Bigger, more powerful engines, larger tires, and additional safety features such as the head and neck safety (HANS) device are just a few items that result from the automobile engineering effort. In this example, the NASCAR team owner (the key stakeholder) wants to win races while also providing the safest possible vehicle for the driver in accordance with the rules, expectations, and constraints established by NASCAR. Based on those stakeholder objectives, NASCAR rules, the specific conditions anticipated on the racetrack, and the strategy for how the team decides to compete, a set of requirements that include performance and safety considerations are defined as part of the engineering process and subsequently, appropriate investments are made to produce a racecar that meets those requirements. While the typical NASCAR race car is more expensive than a family car, the additional expense is justified by the stakeholder mission and business objectives, strategy for competing, and willingness to preserve their most critical asset – the driver.

Knowing the value of your assets and engineering to protect against asset loss and the consequences of such loss – given all types of hazards, threats, and uncertainty – are the focal points of the systems security engineering discipline.

An important objective of systems engineering is to deliver systems deemed *trustworthy*. Trustworthiness is the demonstrated worthiness of a system to be trusted to satisfy given expectations. Claims of trustworthiness are meaningful only to the extent that the needs expressed are accurate, comprehensive, and achievable [Neuman04]. Claims of trustworthiness must include the needs that address adversity. Trustworthiness that is demonstrated only in the

[12] Nomenclature for stages of the system life cycle varies but often includes concept analysis; solution analysis; technology maturation; system design and development; engineering and manufacturing development; production and deployment; training, operations, and support; and retirement and disposal.

absence of adversity fails to account for the concerns of security and is inadequate. The concepts of trust and trustworthiness are discussed in greater detail in Section F.1.

Security is one of several emergent properties of a system. It shares the same issues and challenges in its realization as every other emergent property of the system. Achieving security objectives requires system security activities and considerations to be tightly integrated into all system life cycle stages and the technical and non-technical processes[13] of an engineering effort – thus, the need for trustworthy secure engineering, or *systems security engineering*, as part of demonstrating trustworthiness.

Systems security engineering is an integrative and transdisciplinary approach to enabling the successful and secure realization, use, and retirement of engineered systems using systems, security, and other principles and concepts, as well as scientific, technological, and management methods. Systems security engineering ensures that these principles, concepts, methods, and practices are applied during the entire system life cycle to achieve stakeholder objectives for the protection of assets from all forms of adversity. It also helps to reduce system defects that can lead to vulnerability and, as a result, reduces the effect that adversity can have on the system.

Finally, systems security engineering provides a sufficient base of *evidence* that supports claims or assertions that the desired level of trustworthiness has been achieved – that is, a level of trustworthiness such that the agreed-upon asset protection needs of stakeholders can be satisfied on a continuous basis despite adversity.

As part of a transdisciplinary systems engineering effort to deliver a trustworthy secure system, systems security engineering:

- Works with stakeholders to ensure that security objectives, protection needs/concerns, security requirements, and associated validation methods are defined
- Defines system security requirements[14] and associated verification methods
- Develops security views and viewpoints of the system architecture and design
- Identifies and assesses susceptibilities and vulnerabilities to life cycle hazards and adversities
- Designs proactive and reactive features and functions encompassed within a balanced strategy to control asset loss and associated loss consequences
- Provides security considerations to inform systems engineering efforts with the objective to reduce errors, flaws, and weaknesses that may constitute a security vulnerability
- Performs system security analyses and interprets the results of system security-relevant analyses in support of decision-making for engineering trades and risk management

[13] These stages and processes should possess their own security objectives that support the security objectives.

[14] When the term *system security requirement* is used in this publication, it is important to understand the context in which it is being used. For example, due to the complexity of system security, there are several types and purposes of system security requirements. See Section 2.3.8 and Appendix C.

- Identifies, quantifies, and evaluates the costs and benefits of security features and functions and considerations to inform assessments of alternative solutions, engineering trade-offs, and risk treatment[15] decisions
- Demonstrates through evidence-based reasoning that security and trustworthiness claims for the system have been satisfied
- Leverages multiple security and other specialties to address all feasible solutions

Systems security engineering is considered as a subdiscipline of systems engineering but is not separate; it often overlaps other quality subdisciplines and leverages multiple *specialties* that contribute to systems security engineering activities and tasks. These specialties include computer security; communications security; transmission security; electronic emissions security; anti-tamper protection; physical security; information, software, hardware, and supply chain assurance; and technology specialties such as biometrics and cryptography. Systems security engineering also leverages contributions from other enabling engineering disciplines and specialties[16] to analyze and manage system complexity, dynamicity, interconnectedness, and susceptibility associated with hardware, software, and firmware-based technologies and their development, manufacturing, handling, and distribution throughout the system life cycle.[17] Figure 1 illustrates the relationship among systems engineering, systems security engineering, and contributing security and other specialty engineering areas.

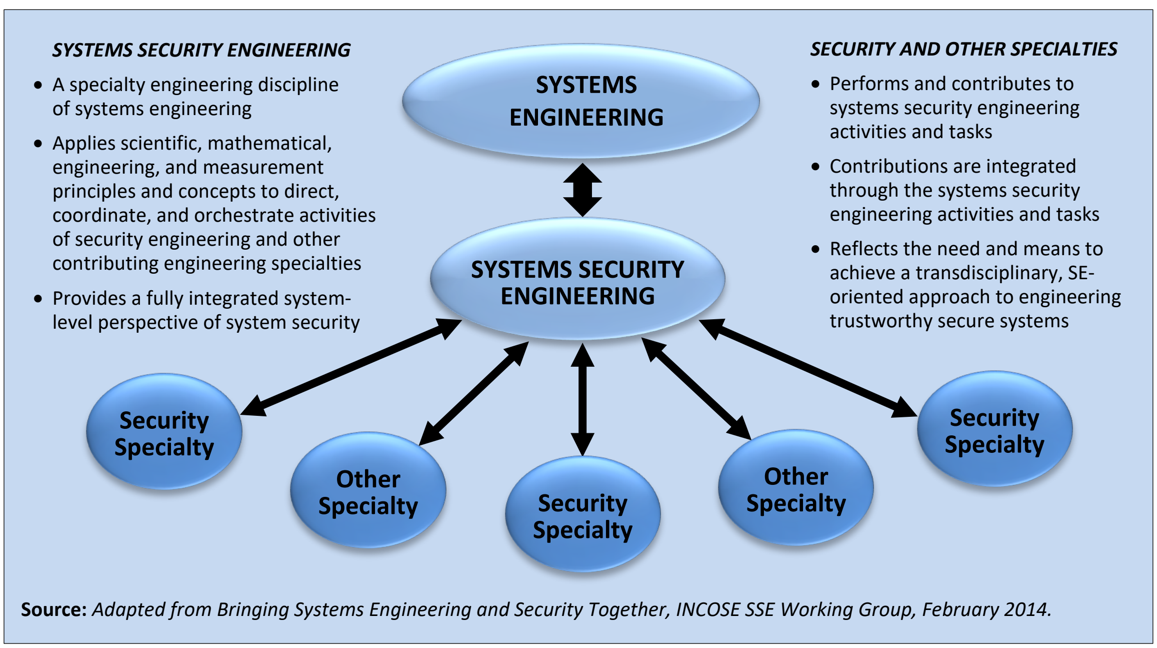


FIGURE 1: SYSTEMS ENGINEERING AND OTHER SPECIALITY ENGINEERING DISCIPLINES

[15] The term *risk treatment* as defined in [ISO 73] is used in [ISO 15288].

[16] Enabling engineering disciplines and specialties include reliability, availability, maintainability (RAM) engineering, software engineering, resilience engineering, and human factors engineering (ergonomics).

[17] This includes assessment of supply chain risk when third-party and reuse considerations are part of the planned system.

2.2 SYSTEM CONCEPTS

Several system concepts are important to understand regarding the engineering of trustworthy secure systems. These include the basic definition of what constitutes a system, the structure of a system, the different categories of systems, and the concept of a system of systems.

2.2.1 Systems and System Structure

A *system* is an arrangement of parts or elements that together exhibit a behavior or meaning that the individual constituents do not.[18] The properties of a system (i.e., attributes, qualities, or characteristics) emerge from the system's constituent parts or elements and their individual properties, as well as the relationships and interactions between and among the parts or elements, the system, and its environment [INCOSE19]. An *engineered system* is a system designed or adapted to interact with an anticipated operational environment to achieve one or more intended purposes while complying with applicable constraints [INCOSE19]. Figure 2 shows the basic structure of a system including its constituent system elements.[19] [20]

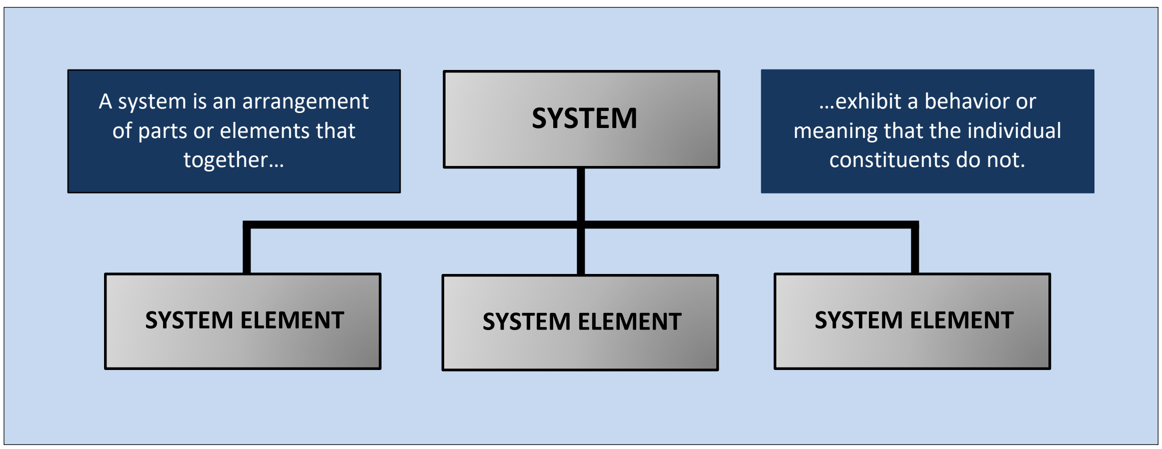


FIGURE 2: BASIC SYSTEM AND SYSTEM ELEMENT RELATIONSHIP

Systems can include:

- Information technology (IT) systems (e.g., general purpose computing systems; command, control, and communication systems; merchandising transaction, inventory, financial management, and personnel systems)
- Internet of Things (IoT) devices (e.g., smart phones, tablets)

[18] A system may be physical (composed of matter and energy), conceptual (composed of information or knowledge), or a combination of both.

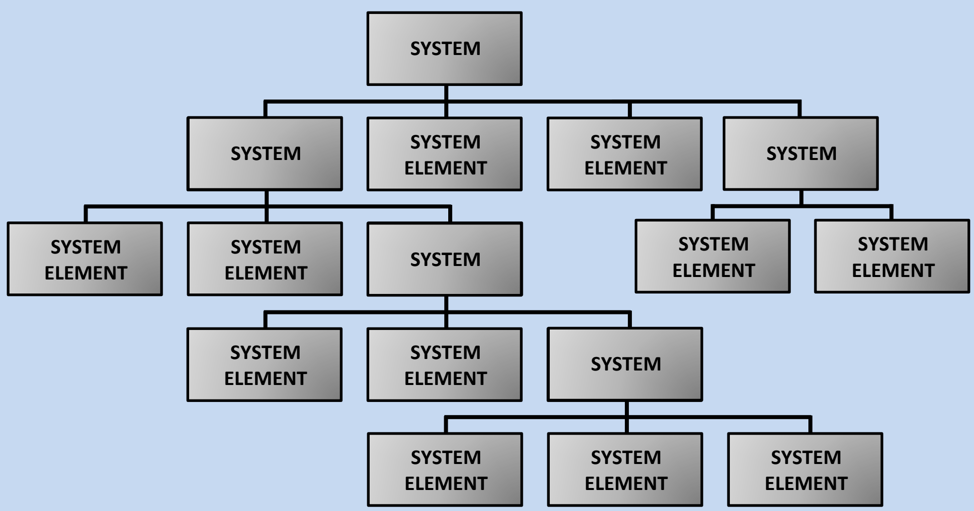
[19] A system element can be a discrete component, product, service, subsystem, system, infrastructure, or enterprise. System elements are implemented by hardware, software, and firmware that perform operations on information or data; physical structures, devices, and components in the environment of operation; and the people, processes, and procedures for operating, sustaining, and supporting the system elements.

[20] In addition to systems with active functions, there are passive systems (physical infrastructure) without such capability that need to exhibit trustworthiness. For example, the interstate highway system employs safety barriers such as Jersey walls (i.e., system elements) that contribute to the trustworthiness of the transportation system.

- Operational technology (OT) systems (e.g., Industrial Control Systems (ICS); Supervisory Control and Data Acquisition (SCADA) and Distributed Control Systems (DCS); Building Management and Building Automation Systems (BMS)/(BAS); weapons systems

The purpose of a system is to deliver a capability or part of a capability, that occurs as a service, function, operation, or a combination thereof. A capability can be delivered by a single system or the emergent combined results of a system of systems. The services, functions, and operations may directly or indirectly interact with, control, or monitor physical, mechanical, hydraulic, or pneumatic devices or other systems or capabilities, or it may provide the ability to create, manipulate, access, transmit, store, or share resources, such as data and information.

As shown in Figure 3, the relationship between system elements can be expressed in many forms (e.g., as hierarchies or networks). A system element may be considered a system (i.e., comprised of other system elements) before a complete set of system elements can be defined. In this manner, the appropriate system life cycle processes are applied recursively to a system of interest to resolve its structure to the point where understandable and manageable system elements can be implemented (i.e., developed, bought, or reused). Note that while the systems and system elements in Figure 3 may imply a hierarchical relationship, many systems are not hierarchical, such as networks and other distributed systems.



Source: [ISO 15288]

FIGURE 3: COMPLEX RELATIONSHIP AMONG SYSTEMS AND SYSTEM ELEMENTS

A *system of systems* is a set of systems and system elements interacting to provide a unique capability that none of the constituent systems can accomplish on its own. The elements of a system of systems are, by definition, systems themselves. A system of systems consists of a number of constituent systems plus any inter-system infrastructure, facilities, and processes necessary to enable the constituent systems to integrate or interoperate [ISO 21841]. Often, a system may be a constituent system in two or more system of systems, further complicating the operational and managerial considerations and stakeholders.

2.2.2 Interfacing, Enabling, and Interoperating Systems

Interfacing systems are systems that interact with the system of interest. Interfacing systems have an interface for exchanging data or information, energy, or other resources with the system of interest. An interfacing system exchanges resources with the system of interest during one or more stages of the system life cycle, such as a system that interfaces for maintenance purposes or a system used to develop the system of interest. The relationships with interfacing systems can be either bi-directional or one way. Interfacing systems have two specific subsets: *enabling systems* and *interoperating systems*.

- **Enabling systems** provide essential services required to create and sustain the system of interest. Examples of enabling systems include software development environments, production systems, training systems, maintenance systems.
- **Interoperating systems** interact with the system of interest for the purpose of jointly performing a function during the utilization and sustainment stages of the system life cycle. Interoperating systems often form a system of systems.

Figure 4 illustrates the relationship between the system of interest and its interfacing systems in both the environment of operation and non-operational (external) environment.

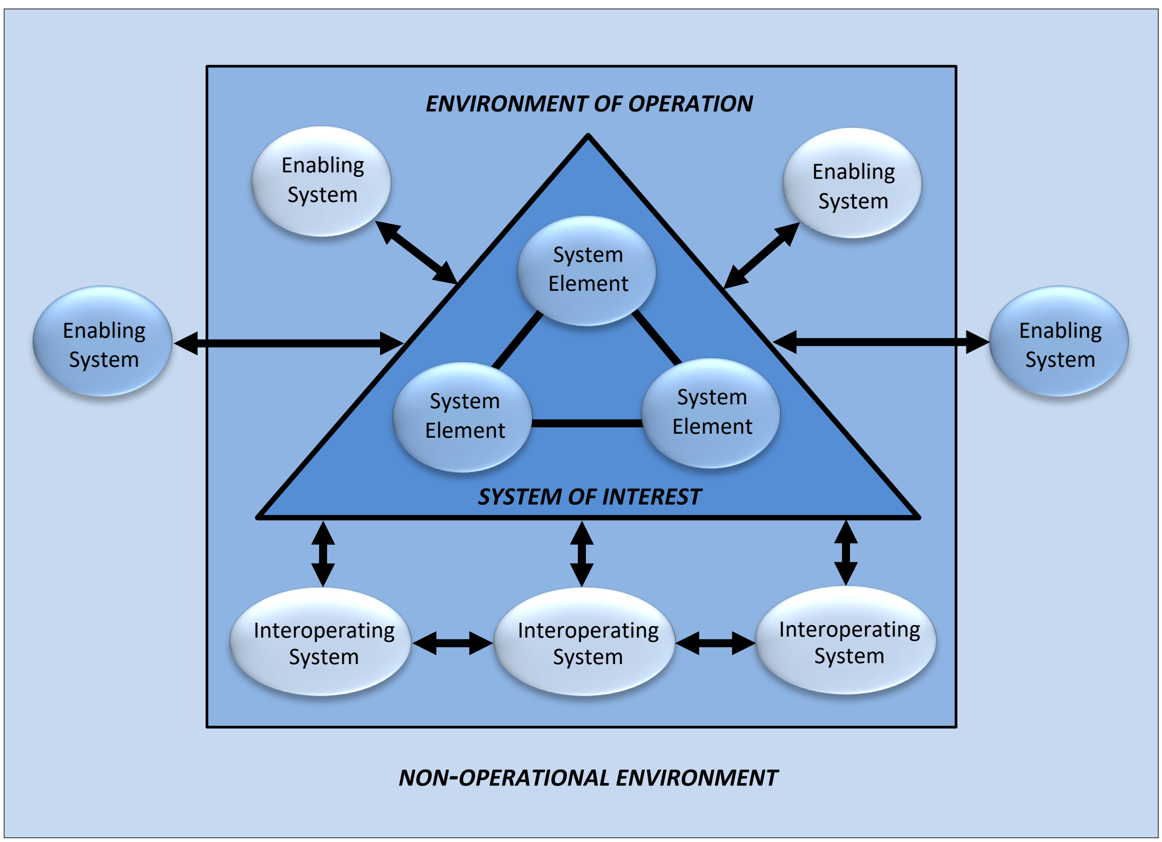


FIGURE 4: SYSTEM OF INTEREST AND INTERFACING SYSTEMS

2.3 SYSTEM SECURITY PERSPECTIVE

Security, as the freedom from the conditions that cause loss of assets with unacceptable consequences, must consider:

- The nature and characteristics of systems (Section 2.3.1) that inform defining conditions
- The nature and concept of loss (Section 2.3.2)
- The concept and adequacy of security (Section 2.3.3)
- The concept of assets (Section 2.3.5) and reasoning about asset loss (Section 2.3.6)
- Protection needs (Section 2.3.7) and various security viewpoints (Section 2.3.8)

2.3.1 The Nature and Character of Systems

The nature and characteristics of systems, their interrelationships with other systems, and their role as part of a system of systems all impact security and efforts to achieve a secure system of interest. The system characteristics that impact system security vary and can include:

- System type, function, and primary purpose[21]
- System technological, mechanical, physical, and human element characteristics
- Modes and states within which the system delivers its functions and services
- Criticality or importance of the system
- Ramifications of the failure of the system to meet its performance expectations, to function correctly, to produce only the intended behaviors and outcomes, and to provide for its own protection (i.e., self-protection)[22]
- System concept for the delivery of a needed capability
- Approach to acquisition of the system, including the assets used in acquisition
- Value and sensitivity of assets entrusted to and used by the system
- Interfaces of the system of interest and systems that interact with the system of interest through those interfaces

Each type of system has differences in terms of its distinct system characteristics and how those characteristics impact the determination of *adequate security* (Section 2.4). For example, a system of systems provides some unique security challenges given the difference in managerial and operational governance compared to other systems. Constituent systems can and do operate independently of one another to fulfill purposes that are distinct from the system of interest. Managerially, the constituent systems are independent and interdependent. The

[21] Some systems are security-purposed systems dedicated to a specific security-oriented function. Such systems may be delivered as a fully independent security capability (e.g., surveillance system), incorporated as a system element within some system (e.g., cryptographic key management system), or attached to a system (e.g., sensor array on an aircraft).

[22] As discussed in Section D.2, a trustworthy secure system must allow only authorized and intended behaviors and outcomes. To the extent possible given constraints and practicality, *self-protection* is a required capability that enables the system to deliver the required stakeholder capabilities while also protecting their assets against loss and the consequences of loss.

managing organizations retain some independence from others and often have their own goals and stakeholders.

2.3.2 The Concept of Loss

Loss is the experience of having an asset[23] taken from one or destroyed or the failure to keep or to continue to have an asset in a desired state or form.[24] The experience of loss is typically the combination of a resultant adverse event or condition and the ramifications, consequences, or impacts of the resultant adverse event or condition. The loss is determined and assessed independent of the causal events and conditions (i.e., the triggering event, such as an error of omission, or the exploitation event, such as an attack). Examples of resultant adverse events or conditions and their ramifications, impacts, or consequences include:

1. **Adverse event or condition:** Data is stolen; it is no longer solely in the possession of the owner or entities authorized by the owner.

   Ramification, impact, or consequence: Market share and competitive advantage is taken away because the data that was stolen provided detailed instructions for a precision machining method that no other company possessed.

2. **Adverse event or condition:** Flat tire on a vehicle; it no longer supports the vehicle weight.

   Ramification, impact, or consequence: One cannot drive the vehicle and needs alternate transportation to get to work, the store, or go on vacation.

While the loss condition or event is negative relative to the intended norm, the effect of the loss can be either neutral/inconsequential or negative/consequential.

Loss may occur because of a single or combination of intentional and unintentional causes, events, and conditions. These may include the authorized or unauthorized use of the system; intentional acts of disruption or subversion; human and machine faults, errors, and failures; human acts of misuse and abuse; and the by-product of emergence, side-effects, and feature interaction. These losses may be inconsequential to the mission or business objectives that are supported by the system, meaning that the mission or business objectives are achieved despite suffering an immediate or eventual loss.

The potential for loss suggests the need for *loss control objectives* that serve as the basis for judgments about the effectiveness of protective measures taken to prevent and limit loss. This includes the resultant adverse events and conditions and the ramifications of those adverse events and conditions. The loss control objectives also serve as the basis to acquire evidence of assurance that the system as designed, built, used, and sustained will adequately protect against loss while achieving its design intent. The loss control objectives reflect an ideal to preserve the characteristics of assets (i.e., state, condition, form, utility) to the extent practicable despite the potential for those characteristics to be changed. The objectives accept uncertainty in the form of limits to what can be done (i.e., not all losses can be avoided) and limits to the effectiveness of what is done (i.e., anything that is done has its scope of effectiveness and set of potential failure modes).

[23] An item of value to one or more stakeholders. See Section 2.3.5.

[24] Adapted from the Merriam Webster definition of loss.

Due to uncertainty, it is not possible to guarantee that some form of loss cannot occur. There is a need to place an emphasis on protection against the effects of loss, including cascading or ripple events (i.e., the immediate effect of a loss is causing some additional unintended or undesired effect or compounding the situation, thereby causing additional losses to occur). Thus, holistically protecting against loss and the unintended or undesired effects of loss considers the full spectrum of possible loss across types of losses and loss effects associated with each asset class. This is important considering that all forms of adversity are not knowable. Therefore, focusing on effect rather than cause when protecting against loss is prudent.

The loss control objectives in Table 1 address the possibilities to control the potential for loss and the effects of loss given the limits of certainty, feasibility, and practicality. Collectively, the loss control objectives encompass the concerns attributed to security and to system safety, survivability, and resilience.

TABLE 1: LOSS CONTROL OBJECTIVES

LOSS CONTROL OBJECTIVE	DISCUSSION
LOSS PREVENTION *(Prevent the loss from occurring)*	• This is the case where a loss is totally avoided. That is, despite the presence of adversity: - The system continues to provide *only* the intended behavior and produces *only* the intended outcomes - The desired properties of the system and assets used by the system are retained - The assets continue to exist • Loss avoidance may be achieved by any combination of: - Preventing or removing the event or events that cause the loss (**the loss never occurs**) - Preventing or removing the condition or conditions that allow the loss to occur (**the loss never occurs**) - Not suffering an adverse effect despite the events or conditions (**the loss never occurs**) • Terms such as *avoid, continue, delay, divert, eliminate, harden, prevent, redirect, remove, tolerate,*[25] *and withstand* are typically used to characterize approaches to achieve this objective such that a loss does not occur despite the system being subjected to adversity
LOSS LIMITATION *(Limit the extent of the loss)*	• This covers cases where a loss can or has occurred, and the extent of loss is to be limited • The extent of loss can be limited in terms of any combination of the following: - Limited dispersion (e.g., migration, propagation, spreading, ripple, domino, or cascading effects) - Limited duration (e.g., milliseconds, minutes, hours, days) - Limited capacity (e.g., diminished utility, delivery of function, service, or capability) - Limited volume (e.g., bits or bytes of data/information) • Decisions to limit the extent of loss may require prioritizing what constitutes acceptable loss across a set of losses, whereby the objective to limit the loss for one asset requires accepting a loss of some other asset • The extreme case of loss limitation is to avoid destruction of the asset • Terms such as *tolerate*, *withstand*, *remove*, *continue*, *constrain*, *stop/halt*, and *restart* fall into this category in the case where the loss occurs and the system can, or enables the ability to, limit the effect of the loss

[25] The term *tolerate* refers to the objective of fault/failure tolerance, whereby adversity in the form of faults, errors, and failures is rendered inconsequential and does not alter or prevent the realization of authorized and intended system behavior and outcomes. That is, the faults, efforts, and failures are tolerated. As used in this publication, tolerate does not refer to a risk acceptance decision.

LOSS CONTROL OBJECTIVE	DISCUSSION
	• Loss recovery and loss delay are two means to limit loss: - *Loss Recovery:* Action is taken by the system or enabled by the system to recover (or allow the recovery of) some or all of its ability to function (i.e., behave, interact, produce outcomes) and to recover assets used by the system (e.g., re-imaging, reloading, or recreating information and data, including software in the system). The restoration of the asset, fully or partially, can limit the dispersion, duration, capacity, or volume of the loss. - *Loss Delay:* The loss event is avoided until the adverse effect is lessened or when a delay enables a more robust response or quicker recovery. • System and environmental conditions may be assumed to result in loss, but measures are taken to limit impacts • Terms such as *contain*, *recover*, *restore*, *reconstitute*, *reconfigure*, and *restart* are typically used to characterize approaches to achieving this objective

2.3.3 The Concept of Security

A system with freedom from those conditions that can cause a loss of assets with unacceptable consequences must provide the intended behaviors and outcomes (e.g., the intended system functionality) and avoid any unintended behaviors and outcomes that constitute a loss. The term *intended* has two cases, both of which must be satisfied:

- **Design intent:** As intended by the design
- **User intent:** As intended by the user

A system that delivers a capability per the design intent but is inconsistent with the user intent constitutes a loss. For example, the loss of control of a vehicle might result from a failure in the vehicle's steering control function (i.e., failure to meet the design intent) or through an attack that takes control away from the driver (i.e., failure to meet the user intent). The primary security objective is to ensure that only the intended behaviors and outcomes occur, both with the system and within the system.[26] Every security need and concern derive from this objective, which is based on the concept of *authorization* for what is and is not allowed.[27] As such, the primary security control objective is the enforcement of constraints in the form of rules for allowed and disallowed behaviors and outcomes. This security control objective – and one of the foundational principles of trustworthy secure design – is *Mediated Access*. If access is not mediated (i.e., controlled though the enforcement of constraints) in accordance with a set of non-conflicting rules, then there is no basis upon which to claim security is achieved.[28]

[26] Behaviors are inclusive of interactions. Interactions of relevance include human-to-machine and machine-to-machine interactions. Human-to-machine interactions are typically transformed into machine-to-machine interactions, whereby a machine element operates on behalf of the human.

[27] An attacker seeks to produce unauthorized behaviors or outcomes. Attackers attempt to accomplish something that they are not authorized to accomplish, even if that behavior or outcome is authorized for some other entity.

[28] The *Reference Monitor Concept* (Section D.4.2) cites three properties of access mediation mechanisms: (1) always invoked, (2) tamper-proof, and (3) evaluatable to substantiate claims of correctness of their implementation. While defined to explicitly address mediated access, the concepts apply equally to any mechanism that enforces constraints on state, behavior, or outcomes.

The rules for mediated access are stated in a set of security policies that reflect or are derived from laws, directives, regulations, life cycle concepts,[29] requirements, or other specifically stated stakeholder objectives. Each security policy includes a *scope of control* that establishes bounds within which the policy applies. Security policy rules are stated in terms of subjects (active entities), objects (passive entities), and the operations that the subject can perform or invoke on the object.[30] The rules govern *subject-to-object* and *subject-to-subject* behaviors and outcomes. The rules for each security policy must be accurate, consistent, compatible, and complete with respect to stakeholder objectives for the scope of control.[31] Inconsistency, incompatibility, or incompleteness in the rules leads to gaps in security protection. It is equally important that the security protection capabilities of the system are aligned with and can achieve the expectations of security policy.

Privileges[32] define the set of allowed and disallowed behavior and outcomes granted to a subject. Privileges are the basis for making mediated access decisions. A restrictive default practice for security policy enforcement is to design the enforcement mechanism to allow only what the policy explicitly allows and to deny everything else. For a system to be deemed trustworthy secure, there must be sufficient confidence that the system is capable of enforcing security policy on a continuous basis for the duration of the time that the security policy is in effect (Appendix F, *Trustworthiness and Assurance*).

Systems engineering must deal with optimizing across multiple objectives that are often in conflict with one another. Often, technologies do not (yet) exist to fully achieve objectives, or they are beyond the constraints of cost and schedule. Therefore, "best effort" is the most that can be practically expected. Given this reality, there is a need to judge best engineering efforts for security.

2.3.4 The Concept of System Security

The definition of security can be interpreted to capture what is meant by a secure system.

> *A secure system is a system that – for all of its identified states, modes, and transitions – ensures that only the authorized intended behaviors and outcomes occur, thereby providing freedom from those conditions, both intentionally/with malice and unintentionally/without malice, that can cause a loss of assets with unacceptable consequences.*

This definition expresses an ideal that captures the three essential aspects of what it means to achieve security:

- Enable the delivery of the required system capability despite intentional and unintentional forms of adversity.
- Enforce constraints to ensure that only the desired behaviors and outcomes associated with the required system capability are realized while satisfying the first aspect.

[29] Life cycle concepts include operation, sustainment, evolution, maintenance, training, startup, and shutdown.

[30] Active entities exhibit behavior (e.g., a process in execution) while passive entities do not (e.g., data, file).

[31] At the highest level of assurance, security policies are formally specified and verified.

[32] Privileges are also referred to as authorizations or rights.

- Enforce constraints based on a set of rules to ensure that only authorized human-to-machine and machine-to-machine interactions and operations are allowed to occur while satisfying the second aspect.

For a system, *adequate security* is an evidence-based determination that achieves and optimizes security performance against all other performance objectives and constraints. Judgments of adequate security are driven by the stakeholder objectives, needs, and concerns associated with the system. Adequate security has two elements:

- Achieve the minimum acceptable threshold of security performance
- Maximize security performance to the extent that any additional increase in security performance results in a degradation of some other aspect of system performance or requires an unacceptable operational commitment

Finally, adequate security is determined based on viewpoint, context, criticality, and priority and may vary across mission or business operational objectives or across the states and modes of the system as it exists (e.g., operation, storage, or transit).[33]

2.3.5 The Concept of Assets

An asset is an item of value. There are many different types of assets. Assets are broadly categorized as either *tangible* or *intangible*. Tangible assets include physical items, such as hardware, computing platforms, or other technology components. Intangible assets include humans, data, firmware, software, capabilities, functions, services, trademarks, intellectual property, copyrights, patents, image, or reputation.[34] Within asset categories, assets can be further identified and described in terms of common asset classes as illustrated in Table 2.

Assets may also be considered as individual items or as an aggregate or group of items that spans asset types or asset classes (e.g., personnel data, fire control function, environmental sensor capability). This publication uses the term *asset of interest* to emphasize and establish bounds on the scope of reasoning for a specific asset, asset type, or asset class.

TABLE 2: COMMON ASSET CLASSES

ASSET CLASS	DESCRIPTION	LOSS PROTECTION CRITERIA
MATERIAL RESOURCES AND INFRASTRUCTURE	This asset class includes physical property (e.g., buildings, facilities, equipment) and physical resources (e.g., water, fuel). It also includes the basic physical and organizational structures and facilities (i.e., infrastructure) needed for an activity or the operation of an enterprise or society.[35] An infrastructure[36] may be comprised of assets in other	*Material resources* are protected from loss if they are not stolen, damaged, or destroyed or are able to function or be used as intended, as needed, and when needed. *Infrastructure* is protected from loss if it meets performance

[33] A system in storage or transit may have expectations to protect critical technologies contained within that system.

[34] Humans are perhaps the most important and valuable of all intangible assets. Safety explicitly considers the human asset, and that same consideration is equally applicable to security.

[35] Adapted from the Merriam Webster and Oxford definitions of *infrastructure*.

[36] There are 16 critical infrastructure sectors whose assets, systems, and networks – whether physical or virtual – are considered so vital to the United States that their incapacitation or destruction would have a debilitating effect on security, national economic security, national public health or safety, or any combination thereof [CISA20].

ASSET CLASS	DESCRIPTION	LOSS PROTECTION CRITERIA
	classes. For example, the National Airspace System (NAS) may be considered infrastructure that itself is a system and contains other elements that are forms of systems and infrastructures, such as Air Traffic Control, navigational aids, weather aids, airports, and the aircraft that maneuver within the NAS.	expectations while delivering only the authorized and intended capability and producing only the authorized and intended outcomes.
SYSTEM CAPABILITY	This asset class is the set of capabilities or services provided by the system. Generally, system capability is determined by: (1) the nature of the system (e.g., entertainment, vehicular, medical, financial, industrial, or recreational); and (2) the use of the system to achieve mission or business objectives.	*System capability* is protected from loss if the system meets its performance expectations while delivering only the authorized and intended capability and producing only the authorized and intended outcomes.
HUMAN RESOURCES	This asset class includes personnel who are part of the system and personnel who are directly or indirectly involved with or affected by the system. The consequences of loss associated with the system may significantly change the importance of this asset class (e.g., the effect on personnel due to a failure of a guidance system in an aircraft is significantly different from the effect on personnel due to the breach of a system that compromises individual credit card information).	*Human resources* are protected from loss if they are not injured, suffer illness, or killed.
INTELLECTUAL PROPERTY[37]	This asset class includes trade secrets, recipes, technology,[38] and other items that constitute an advantage over competitors. The advantage is domain-specific and may be referred to as a competitive advantage, technological advantage, or combative advantage.	*Intellectual property* is protected from loss if it is not stolen, corrupted, destroyed, copied, substituted in an unauthorized manner, or reverse-engineered in an unauthorized manner.
DATA AND INFORMATION	This asset class includes all types of data and information (aggregations of data) and all encodings and representations of data and information (e.g., digital, optical, audio, visual). There are general sensitivity classes of data and information that do not fall within the above categories, such as classified information, Controlled Unclassified Information (CUI), and unclassified data and information.	*Data and information* are protected from loss due to unauthorized alteration, exfiltration, infiltration, and destruction.
DERIVATIVE NON-TANGIBLES	This asset class is comprised of derivative, non-tangible assets, such as image, reputation, and trust. These assets are defined, assessed, and affected – positively and negatively – by the success or failure to provide adequate protection for assets in the other classes.	*Non-tangible assets* are protected from loss by ensuring the adequate protection of assets in the other classes.

[37] The term *intellectual property* is defined as an output of a creative human thought process that has some intellectual or informational value [ISO 24765]. Examples include microcomputer design and computer programs.

[38] The term *technology* is defined as the application of scientific knowledge, tools, techniques, crafts, systems, or methods of organization to solve a problem or achieve an objective [ISO 16290].

The *valuation* of an asset is a key input in decision-making about investments to protect an asset. The valuation determination is made by stakeholders. For those cases where an asset is associated with multiple stakeholders, there may be differing, contradictory, competing, or conflicting concerns about the valuation of the asset. These differences are addressed as part of discussions that resolve differences associated with agreements on needs, expectations, and requirements. The valuation of an asset may be influenced by a variety of factors that include the cost (i.e., monetary, time, material, human resources) to develop or acquire, the cost to maintain, the cost to repair or replace, the cost if the asset is not repairable or replaceable, and the importance of completing an objective.[39]

2.3.6 Reasoning About Asset Loss

The elements of a structured approach for reasoning about assets and assets loss are shown in Figure 5. The elements provide a comprehensive basis for decision-making about assets and asset loss to determine the objectives for a secure system, optimize the protection capability of the system, and make judgments on the suitability and effectiveness of the implemented protections.[40] Each of these elements is discussed in greater detail below.

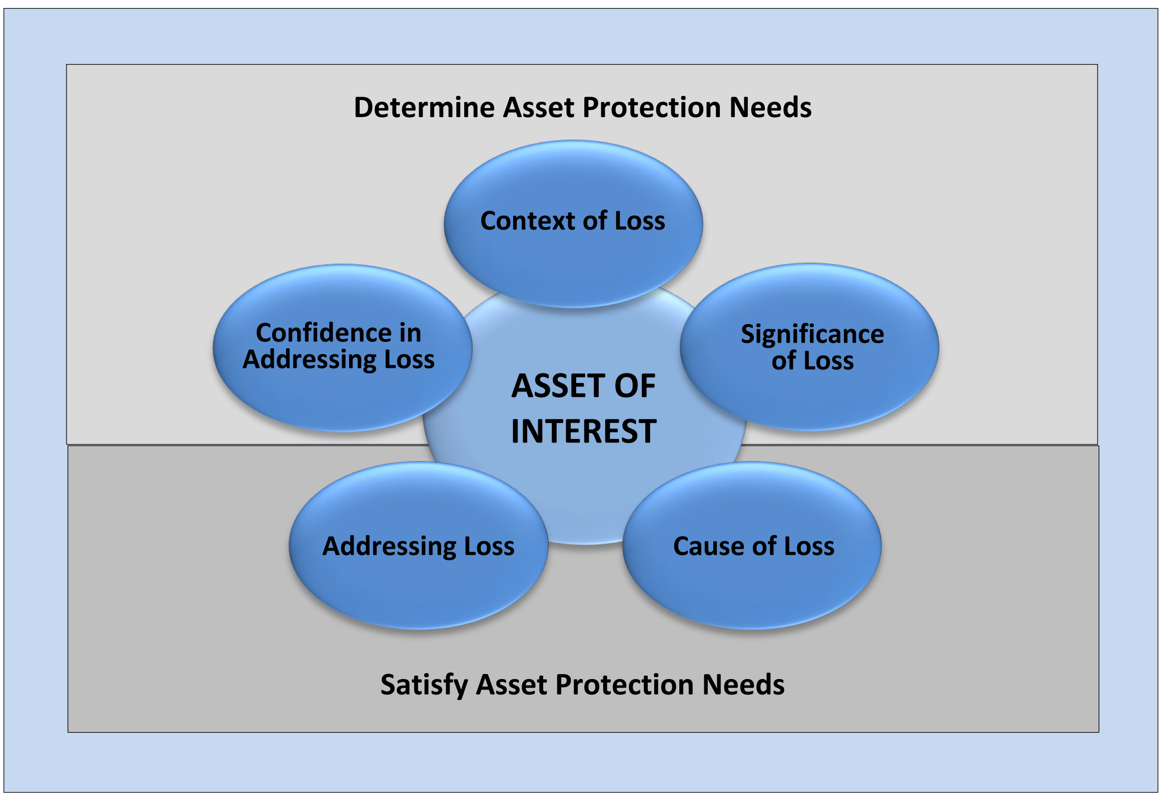


FIGURE 5: REASONING ABOUT ASSET PROTECTION

[39] The Department of Defense's *Mission Engineering Guide* [DOD 2020] relates asset protection to mission by using a mission objective of preserving a return on investment (ROI). Life, material, technological advantage, or other unintentional losses that occur while executing a mission may be considered a poor return on investment.

[40] The application of the asset reasoning approach works equally to reason about assets in terms of mission (i.e., mission-driven asset reasoning), organization (i.e., organization-driven asset reasoning), and enterprise (i.e., enterprise-driven asset reasoning).

The elements are grouped into two objectives to facilitate reasoning about the *asset of interest*:

- **OBJECTIVE 1:** *Determine* asset protection needs
  - **Context of Loss:** The scope and criteria that bounds reasoning about asset loss
  - **Significance of Loss:** The effect of asset loss (or adverse impact) based upon its valuation
  - **Confidence in Addressing Loss:** The assurance to be achieved based on claims-driven and evidence-based arguments about the effectiveness of what is done to address potential and actual loss
- **OBJECTIVE 2:** *Satisfy* asset protection needs
  - **Cause of Loss:** The events, conditions, or circumstances that describe what has happened before and what can happen in the future and that constitute the potential for loss to occur
  - **Addressing Loss:** The various actions taken to exercise control over loss to the extent practicable. The control objectives are to prevent loss from occurring and to limit the extent and duration for those losses that do occur. Limiting loss includes recovery from loss to the extent practicable.

The *asset of interest* is the asset class, asset type, or individual asset being addressed. Reasoning about loss is based on the asset of interest. Distinguishing the asset of interest from all other assets provides clarity in the interpretation of loss for the asset of interest and the associated judgments of suitability and effectiveness of protections employed. A focus on a specific asset class, type, or discrete element also enables precise traceability to requirements that support the analysis needed to determine the protection-relevant impact of changes to requirements.

The *context of loss* sets the boundary, scope, and time frame for the reasoning, analyses, assessments, and conclusions about the asset of interest. The context of loss also provides a basis to relate and trace asset dependencies and interactions and to group assets for protection. The context of loss time frame is particularly important because the asset of interest has a life cycle[41] that is different from the system of interest.[42] For example, the asset of interest may be created, configured, or modified outside of the scope of control of the system of interest yet be within the scope of the engineering effort. The asset of interest, once within the scope of control of the system of interest, may have differing protection needs associated with the state or mode of the system (e.g., the system operational mode protection may differ from the system training mode). Additionally, system life cycle assets (Section 2.3.8) may exist only within a development or production system and their associated supporting environments. The effect of the loss for these assets may transfer to a loss associated with the system of interest. Therefore, the context of loss includes the life cycle of the asset, the state and mode of the system, and other time-based periods or characteristics during which loss is addressed.

[41] The lifetime of an asset may be different from the lifetime of the system. Assets may predate the system and may persist after the system's retirement from use. The significance of the loss of an asset can have ramifications that are independent of the system, system function, and business and mission objectives.

[42] The asset life cycle is the same as the system life cycle when the asset of interest is the system of interest. The asset life cycle may be the same or shorter than the system life cycle for those assets created by the system of interest and only required while the system of interest is operating.

TIMEFRAME OF LOSS – AN EXAMPLE

A financial portfolio (an asset or collection of assets) with specific investment objectives and risk acceptance considerations may be created by a financial advisor for a client, funded by the client, and subsequently managed using multiple systems across one or more institutional investment firms throughout the portfolio's life cycle. Each asset of interest within the portfolio may have differing protection needs at different times depending on the type of asset, market conditions, regulatory jurisdiction, risk position, and other asset management factors that are imposed on the system.

The *significance of loss* is the adverse effect on the asset of interest or the resultant adverse effect associated with the asset. The significance of loss is best described as an experience that is to be avoided, thereby warranting an investment to protect against it occurring and to minimize the extent of the adverse effect should it occur. The significance of loss is determined and assessed as an effects-based judgment. That is, it is determined without any consideration of how or why the loss occurs, the probability or likelihood of the loss occurring, and any intent or the absence of intent related to the loss.[43]

The *consequence of loss* simply answers the following question: "What are the ramifications, effects, and problems that result from suffering a loss of the asset of interest?" The significance of loss requires clarity in what loss means for the asset of interest. Examples of terms used to describe asset loss include ability, accessibility, accuracy, assurance, advantage (technological, competitive, combatant), capability, control, correctness, existence, investment, ownership, performance, possession, precision, quality, satisfaction, and time.

Confidence in addressing loss ensures that protections have a body of objective evidence that demonstrates the effectiveness, sufficiency, and suitability of protective measures to satisfy asset protection needs. Confidence in addressing loss is cumulative. It begins with determining the loss concerns for the asset of interest and continuously builds as those concerns are better understood and addressed across the context of loss, the consequence of loss, the causes of loss, and how loss is addressed. The evidence basis that provides confidence is informed by verification and validation activities that occur throughout the life cycles of the assets and the system, including requirements elicitation and analysis. A key informing element to those activities is to ensure that the results contribute to the confidence sought.

The *cause of loss*[44] is the individual or combination of events, conditions, and circumstances that result in some form of loss of an asset. The causes of asset loss constitute a continuum that

[43] Determining the consequence of loss is not a determination of risk.

[44] Many terms are used to describe the cause of asset loss. Some of these terms are specific to a community of interest or specialty field, while others span communities and specialties. There are also cases where the same term may be used differently across communities and specialty fields (e.g., the term *threat* has varying interpretations across communities, such as physical security, cybersecurity, commerce, law enforcement, industry, military combat operations, and military intelligence). The terms typically used as a synonym for the cause of asset loss include attack, breach, compromise, hazard, mishap, threat, violation, and vulnerability.

includes intentional, unintentional, accidental, incidental, misuse, abuse, error, defect, fault, weakness, and failure events and conditions. This continuum spans all human-based, machine-based, physical-based, and nature-based drivers of loss. The following considerations apply to reasoning about the causes of loss:

- Single events and conditions that alone can produce the loss
- Combinations, sequences, and aggregate events and conditions
- Events and conditions that are desirable, intended, and even planned yet produce unanticipated, unforeseen, and unpredictable results
- Cascading and ripple events and conditions

Finally, the causes of asset loss answer the questions: "How can loss occur, and how has loss occurred in the past?" The purpose of determining how loss can occur does not ask the question "What is likely or probable to happen?"[45]

SIGNIFICANCE OF LOSS – AN EXAMPLE

The significance of loss due to a flat tire is determined and assessed without consideration of how or why the tire became flat (e.g., puncture, manufacturing defect, impact with curb or other object) and without any consideration of malicious intent (e.g., tire cut, valve stem loosened). Regardless of how or why the tire became flat, the significance of loss remains the same (e.g., loss of control if the vehicle is moving, inability to drive if the vehicle is stationary, time lost to replace or repair the tire to make the vehicle operable). The significance of loss due to a flat tire includes the inability to steer the vehicle, and the resultant adverse effect may be to impact some other object (i.e., a crash). The adverse effect of the loss of steering (loss of control) is specific, while the adverse effect of a crash is general (many other circumstances may result in a crash without any loss of the ability to steer the vehicle).

Addressing loss occurs through the protective measures that enforce constraints to ensure that only authorized and intended behaviors and outcomes of the system occur. These include:

- Protective measures provided by the *machine* portion of the system (i.e., the system architecture and design, the use of engineered features and devices within the architecture and design)
- Protective measures provided by the *human* portion of the system (i.e., personnel, procedures, practices, the use of tools to support the human as a system element, and the human role in designing and building the machine part of the system)
- Protective measures provided by the *physical environment* (i.e., facility access points, controlled access areas, physical monitoring, environmental controls, and fire suppression)

[45] This point distinguishes analysis of what can happen from a risk assessment that determines probability greater than zero and less than one that the adverse event will happen.

The terminology used to describe means and methods includes configurations, controls, countermeasures, features, inhibits, mechanisms, overrides, practices, procedures, processes, safeguards, and techniques. These may be applied in accordance with governing policies, regulations, laws, practices, standards, and techniques.

2.3.7 Protection Needs

Stakeholders have a need to achieve their mission or business objectives in a secure manner that preserves assets and limits the extent of asset loss. Asset protection must be continuous, thereby making it possible for stakeholders to have a realistic expectation of continuous success in the ability of their systems to support and achieve their objectives.

The scope and expectations for the protection of assets is foundational to achieving the design intent for a trustworthy secure system. Protection needs typically correlate to the severity of consequences associated with the loss of an asset. The protection needs are determined from all needs, concerns, priorities, and constraints to protect and preserve stakeholder and system assets. There are two perspectives for protection needs: (1) the *stakeholder* perspective; and (2) the *system* perspective. Figure 6 illustrates the key input sources used to define protection needs and the outputs derived from the specification of those needs.

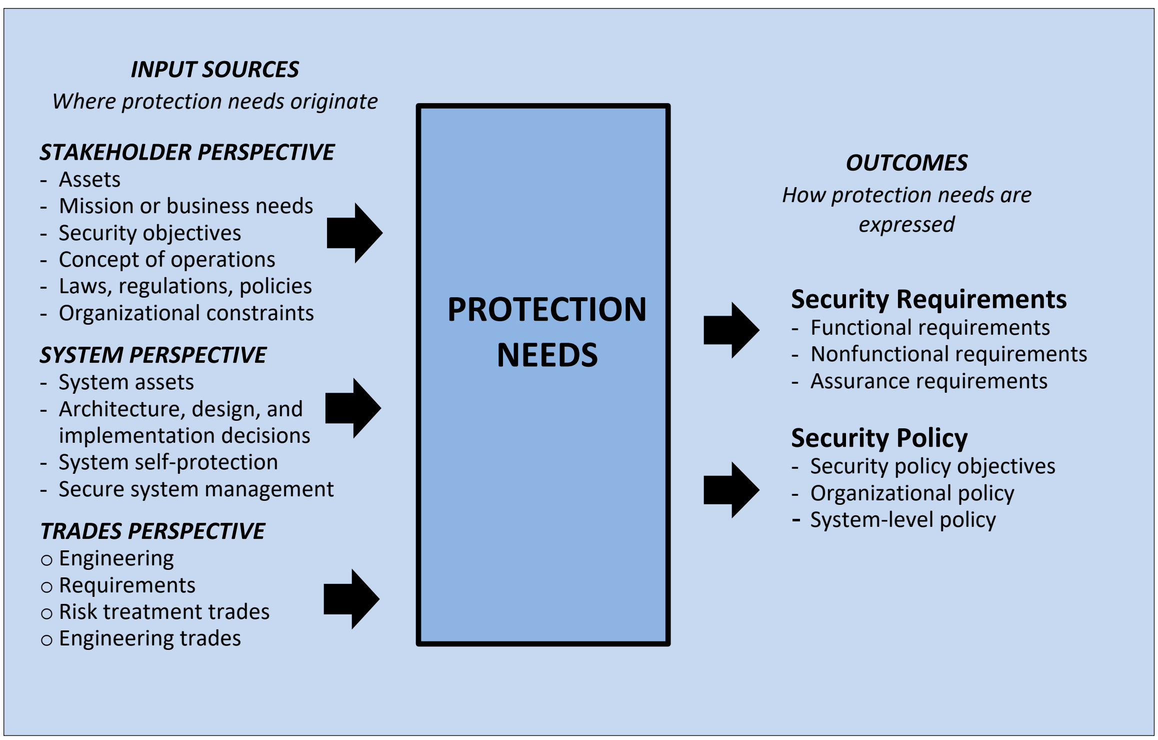


FIGURE 6: DEFINING PROTECTION NEEDS

The stakeholder perspective is based on the assets that belong to stakeholders. Therefore, those stakeholders determine the protection needs. The system perspective is based on the assets necessary for the system to function. These assets are determined by system design decisions and the criticality and priority[46] of the asset in providing or supporting the functions of the system. Stakeholders are typically unaware of the existence of system assets and are not able to make decisions about the protection needs for system assets. The protection of system assets is an element of trustworthy secure system design.

The purpose of establishing the *need for protection* is to decide what assets to protect and to determine the priority given to such protection. This can be accomplished without considering a cause or condition against which to protect. As shown in Figure 7, the need for protection is derived from the relationship among the asset of interest, context of loss, type of loss, and the consequences of loss. This approach establishes the need for protection that, once validated by stakeholders across all assets of interest, provides the basis for developing security objectives and requirements.[47]

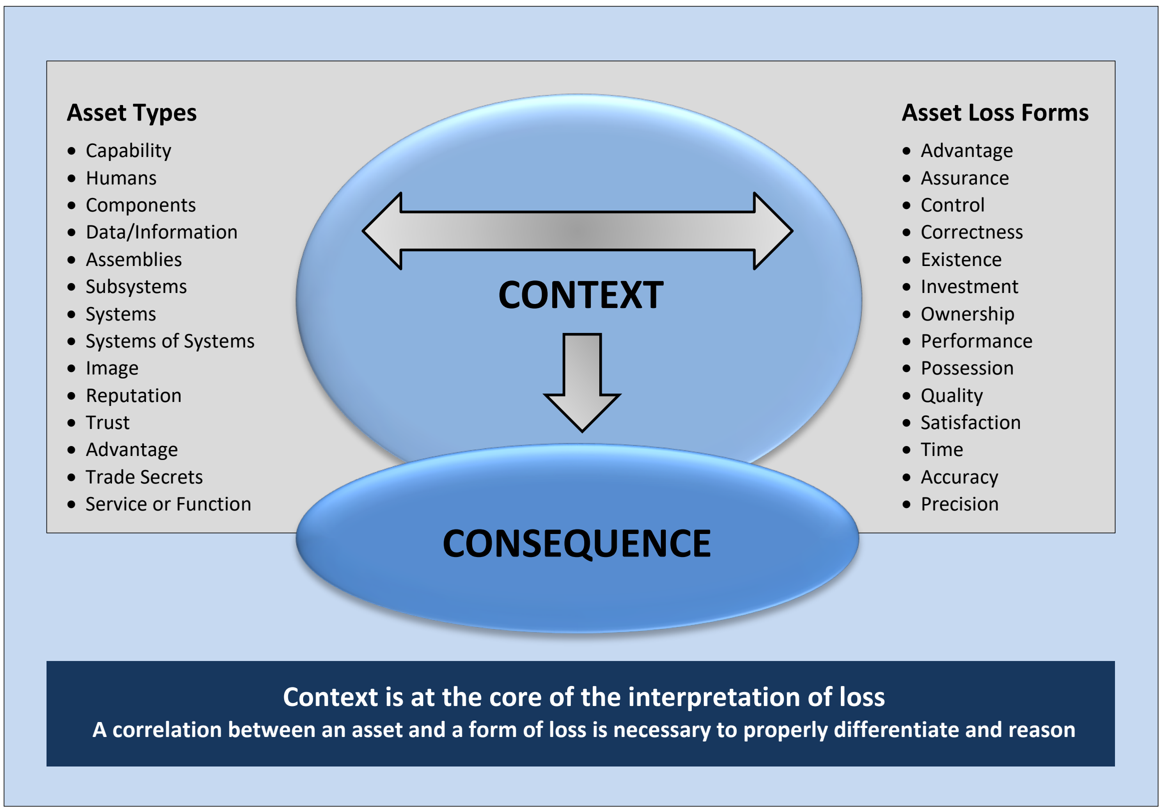


FIGURE 7: RELATIONSHIP AMONG ASSET, LOSS, AND CONSEQUENCE

Summarizing, the following considerations impact the identification of protection needs:

- Assets have different classes and types

[46] Criticality and priority based on asset valuation is typically used in decisions on protection needs. An asset with higher criticality and priority would take precedence in providing protection should there be constraints that require making choices between the overall protection needs (Section 2.3.7).

[47] Requirements provide a formal and clear expression of the needs, concerns, priorities, and constraints to be satisfied for system function, operation, and maintenance. Each requirement is accompanied by verification methods for demonstrating that the requirement is satisfied. Requirements must be accurate, unambiguous, comprehensive, evaluatable, and achievable.

- Assets are associated with stakeholders and the system
  - Some assets are associated with stakeholders (i.e., stakeholder assets) and have a purpose, use, and existence that is independent of the system being designed
  - Some assets are associated with the system, are dependent on characteristics of the system design and behavior, and are typically unknown to stakeholders
- Loss interpretation is dual-faceted
  - The effect on the asset of interest
  - The effect on those who value the asset of interest
- Loss interpretation is temporal and state-based
  - Spans a continuum within and across asset types and classes
  - May change across the life cycle of the asset and the state in which the asset exists or is utilized
- Asset-based judgments are subjective
  - Asset valuation
  - Asset loss ramifications
  - Asset protection suitability, effectiveness, and dependability

ASSET-BASED PROTECTION – ENGINEERING FOR SUCCESS

Don't focus on what is *likely* to happen. Instead, focus on what *can* happen, and be prepared. That is what systems security engineering means by adopting a proactive and reactive strategy (Section D.2) in the form of a *concept of secure function* that addresses the spectrum of asset loss and associated consequences. This means proactively planning and designing to prevent the loss of an asset that you are not willing to accept, to be able to minimize the consequences should such a loss occur, and to be in an informed position to reactively recover from the loss when it does happen.

Protection needs are continuously reassessed and adjusted as variances, changes, and trades occur throughout the system life cycle. These include the maturation of the system design and life cycle concepts, improved understanding of the operational environment (e.g., a more thorough understanding of adversities), and changes in understanding the consequences of asset loss. Revisiting protection needs is a necessary part of the iterative nature of systems engineering and with it, systems security engineering—necessary to ensure completeness in understanding the problem space, exploring all feasible solutions, and engineering a trustworthy secure system.

2.3.8 System Security Viewpoints

The three predominant views of system security that support trustworthy secure design considerations for any system type, intended use, and consequence of system failure are *system function*, *security function*, and *life cycle assets*.

Every system is delivered to satisfy stakeholder capability needs. These needs constitute the *system functions.* Securely satisfying stakeholder capability needs requires the enforcement of security-driven constraints that combine with the overall design of the system. The security-driven constraints are provided by the *security functions* of the system. These constraints focus on the avoidance (i.e., preferred outcome), reduction, and tolerance of susceptibilities, defects, weaknesses, and flaws in the system that may constitute a vulnerability that can be exploited or triggered. These vulnerabilities may be within the system's structure or within its behaviors, including vulnerabilities that counter, defeat, or minimize the ability of the security functions to effectively satisfy their design intent. Thus, the constraints also enable the synthesis of security functions into the system in a non-conflicting manner.

Security functions are those functions of the system whose sole purpose is to satisfy objectives to control asset loss (including the loss of intended behavior and outcomes) and the associated consequences. Security functions are realized by the employment of engineered features and devices, generally referred to as controls, countermeasures, features, inhibits, mechanisms, overrides, safeguards, security controls, or security services. Security functions have both *passive* and *active* aspects:

- Passive aspects of security functions do not exhibit behavior. They include the system architecture and design elements. The passive aspects are part of the system structure and require consideration in the architecture of the system. For example, the functional architecture may segment system functions (including security functions) into different subsystems, reducing the possibility of interference among functions as well as limiting the propagation of erroneous behavior. Passive aspects inherently reduce the susceptibility of the system to exposure, hazard, and vulnerability, thereby limiting if not eliminating the potential for loss scenarios. The employment of passive aspects generally enables greater confidence in the protection capability of the system.
- Active aspects of security functions exhibit behavior (i.e., are functional in nature). The active aspects are employed or allocated within the system architecture, have a specific design, and have capabilities and limitations that affect their suitability and effectiveness relative to their intended use.

Life cycle assets are those assets that are associated with the system but are not engineered into the system or delivered with the system. Their association with the system means that they can be the direct cause of loss or a conduit/means through which a loss can occur. Life cycle assets have several types:

- Systems that interact with the system of interest in its environment of operation, including conceptual systems (Section 2.2.1)
- Intellectual property in various forms, including proprietary algorithms, technologies, and technology solutions
- Data and information associated with the system

- Developmental, manufacturing, fabrication, and production capabilities, systems, and environment systems and capabilities used to utilize, operate, and sustain the system[48]

> **SECURITY FUNCTIONS – PASSIVE AND ACTIVE ASPECTS**
>
> As discussed in Section D.3, *passive* security functions (i.e., structure) have certain advantages over *active* security functions due to their greater potential for assurance in achieving objectives. However, both types of functions are needed and are complementary (e.g., a good structure can increase the effectiveness of an active function). Passive and active aspects of security functions factor into trades, as discussed in Section D.4.4. Active security functions also require additional hardware or loads on existing hardware, increasing demands for size, weight, and power (SWaP) and making active functions a challenge for SWaP-restricted systems (e.g., satellites).

2.4 DEMONSTRATING SYSTEM SECURITY

The system security definition (i.e., freedom from those conditions that can cause a loss of assets with unacceptable consequences) brings an inherently context-sensitive and subjective nature to assertions or expectations about the system security objectives and the determination that those objectives have been achieved. The context sensitivity and subjectivity occur because no individual stakeholder can speak on behalf of all stakeholders regarding the ramifications or effects of the loss of stakeholder and system assets throughout the system life cycle.

Moreover, system security, as an *emergent property* of the system, is an outcome that results from and is assessed in terms of the composed results of the system element parts. System security is not determined relative to an assessment of any one part or collection of parts without considering the whole.[49] Therefore, the requirements and associated verification and validation methods, while necessary, are not sufficient as the basis to deem a system secure. The requirements and the life cycle concepts informing those requirements must be shown to be comprehensive and sufficient. What is necessary is the means to address the emergent property of security across the subjective and often contradicting, competing, and conflicting needs and beliefs of stakeholders and to do so with a level of confidence that is commensurate with the asset loss consequences that are to be addressed (Appendix F).

This is achieved through the type of diligent and targeted reasoning that forms the basis of assurance cases (Appendix F). The reasoning considers the system needs and capabilities, contributing system quantitative and qualitative factors, and how these capabilities and factors compose in the context of system security to produce an evidentiary base upon which analyses

[48] Examples include software and hardware development tools and suites; modeling and simulation environments and tools; maintenance and diagnostics devices, components, and suites; simulators and test-case scenario generators; and training systems. While these assets are not necessarily within the scope of engineering the system of interest, behaviors and outcomes of these systems have security implications that must be addressed in the secure design of the system of interest. The behaviors and outcomes to consider include how they might directly or indirectly enable, interface, interact, and interoperate with the system of interest.

[49] An individual function or mechanism can be verified and validated for correctness and for its specific quality and performance attributes. Those results inform the determination of system security but do not substitute for them.

are conducted. These analyses, in turn, support substantiated and reasoned conclusions that serve as the basis for consensus among stakeholders.[50] The ultimate objective is to be able to claim with sufficient confidence or assurance that the system is *adequately secure* relative to all stakeholders' objectives, concerns, and associated constraints and to do so in a manner that is meaningful to stakeholders and that can be recorded, traced, and evolved as variances occur throughout the system life cycle. There will never be absolute assurance, however, because of the asymmetry in system security – that is, things can be declared insecure by observation, but there is no observation that allows one to declare an arbitrary system secure [Herley16].

The scope of conditions relevant to security is specific to the stakeholder needs to be met by the system. This is also the case for the level of security to be considered acceptable. Absolute security is not expected to be attainable. Rather, a sufficient level of security is needed to fulfill protection need priorities. To be *adequately secure*,[51] the system:

- Is assessed to meet minimum tolerable levels of security, as determined by analysis, experience, or a combination of both. Below such levels the system is considered insecure.
- Is as secure as reasonably practicable (ASARP); that is, incremental improvement in security would require an intolerable or disproportionate deterioration of meeting other system objectives such as those for system performance, would violate system constraints, or would require unacceptable concessions such as an unacceptable change in the way operations are performed.

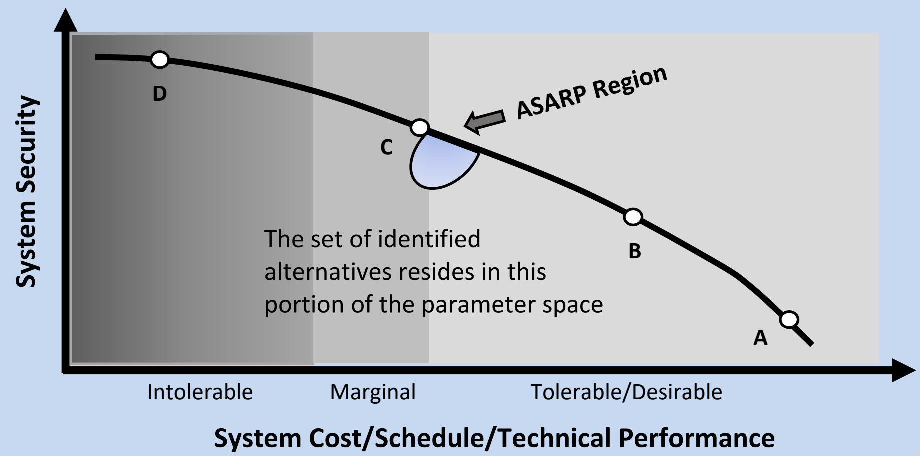
An adequately secure system does not necessarily preclude all of the conditions that can lead to undesirable consequences. The minimum tolerable levels of security and interpretations of "as secure as reasonably practicable" may not be fixed over the life of a system. The information gathered while the system is in use and the lessons learned may inform candidate modifications that raise the bar on either or both. Figure 8 illustrates the tradeoffs between system security and the cost, schedule, and technical performance of the system.

ADEQUATE SECURITY

No system can provide *absolute* security due to the limits of human certainty, the uncertainty that exists in the life cycle of every system, and the constraints of cost, schedule, performance, feasibility, and practicality. As such, trade-offs made routinely across contradictory, competing, and conflicting needs and constraints are optimized to achieve *adequate* security, which reflects a decision made by stakeholders.

[50] System security requirements development must be iterative with the involvement of stakeholders, regardless of the life cycle model used. Such development spans several life cycle processes as described in Chapter Three. The iterative development of system security requirements is necessary to address the evolution and maturation of the system as it proceeds from concept to design and, subsequently, to its "as-built" forms.

[51] The concept of *adequately secure* is an adaptation of the concept of *adequately safe* from [NASA14].



- Large increases in system security can be achieved by addressing basic security issues. Little cost, schedule, or technical impact.
- Basic security issues have been addressed but significant security can still be "bought" without failing to meet cost, schedule, or technical performance requirements.
- Limit of ASARP regime has been reached but significant increases in security can be "bought" without exceeding tolerable limits of cost, schedule, or technical performance requirements.
- Limit of achievable security has been met. Increased security cannot be "bought" at any cost.

Adapted from [NASA14].

FIGURE 8: SYSTEM SECURITY AND COST/SCHEDULE/TECHNICAL PERFORMANCE

2.5 SYSTEMS SECURITY ENGINEERING FRAMEWORK

The *systems security engineering framework* [McEvilley15] provides a conceptual view of the key contexts within which systems security engineering activities are conducted. The framework defines, bounds, and focuses the systems security engineering technical and non-technical activities and tasks towards the achievement of stakeholder *security objectives* and presents a coherent, well-formed, evidence-based case that those objectives have been achieved.[52] The framework is independent of system type and engineering or acquisition process model and is not to be interpreted as a sequence of flows or process steps but rather as a set of interacting contexts, each with its own checks and balances. The systems security engineering framework emphasizes an integrated, holistic security perspective across all stages of the system life cycle and is applied to satisfy the milestone objectives of each life cycle stage.

The framework defines three contexts for conducting systems security engineering activities: (1) the *problem* context, (2) the *solution* context, (3) and the *trustworthiness* context. Establishing the three contexts helps to ensure that the engineering of a system is driven by a sufficiently complete understanding of the problem. This understanding is described in a set of stakeholder security objectives that reflect protection needs and security concerns instead of by security solutions brought forth in the absence of consideration of the entire problem space and its associated constraints. Moreover, there is explicit focus and a set of activities to demonstrate

[52] Adapted from [NASA11].

the worthiness of the solution in providing adequate security across competing and often conflicting constraints. While the framework appears to follow a *sequential* execution across the three contexts, it is actually implemented in an *iterative* manner within the stages of the system life cycle and is guided and informed by system analyses (Section 3.4.6). The transition from stage to stage in the life cycle is controlled by decision gates. Iteration facilitates refinement of the problem statement, proposed solutions, and trustworthiness objectives.

Figure 9 illustrates the systems security engineering framework and its key components.

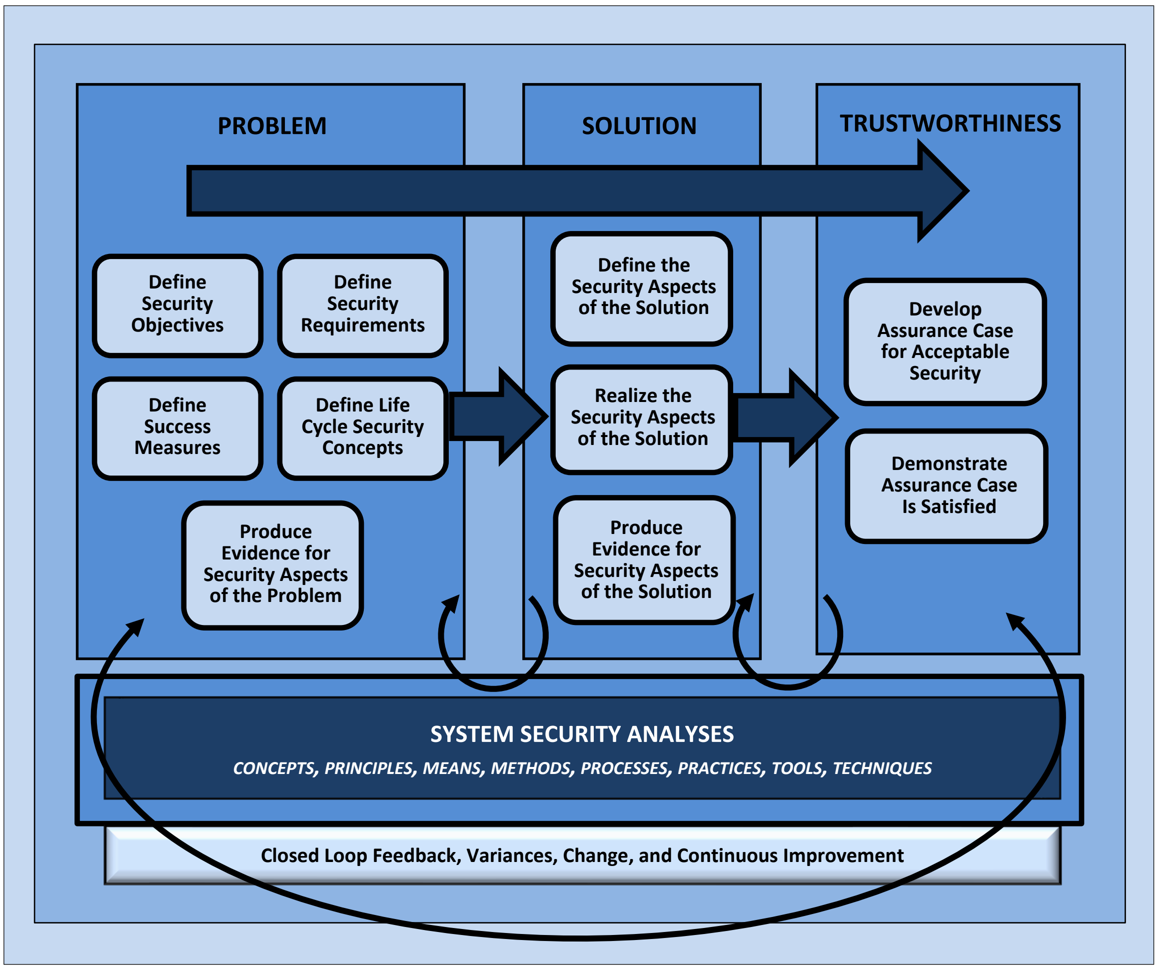


FIGURE 9: SYSTEMS SECURITY ENGINEERING FRAMEWORK

The contexts of the systems security engineering framework share a common foundational base of *system security analyses,* including *system analyses* with security interpretations of resulting data. System security analyses produce data to support engineering and stakeholder decision-making. Such analyses are differentiated for application within the problem, solution, and trustworthiness contexts and routinely employ concepts, principles, means, methods, processes, practices, tools, and techniques. System security analyses:

- Provide relevant data and technical interpretations of system issues from the system security perspective

- Are differentiated in their application to align with the scope and objectives of where they are applied within the systems security engineering framework
- Are performed with a level of fidelity, rigor, and formality to produce data with a level of confidence that matches the assurance required by the stakeholders and engineering team (see Appendix F)

System security analyses address important topic areas related to systems security engineering. These areas include architecture, assurance, behavior, cost, criticality, design, effectiveness, emergence, exposure, fit-for-purpose, life cycle concepts, penetration resistance, performance (including security performance), protection needs, privacy, requirements, resilience, risk, strength of function, security objectives, threats, trades, uncertainty, vulnerability, verification, and validation.

The systems security engineering framework includes a *closed loop feedback* for interactions among and between the three framework contexts and the requisite system security analyses to continuously identify and address variances as they are introduced into the engineering effort. The feedback loop also helps to achieve continuous process improvement for the system, including viewing the outputs of one life cycle phase (i.e., the "solution" to the phase) as the inputs to the next phase (i.e., the "problem" for the next phase).

2.5.1 The Problem Context

The *problem context* defines the basis for an acceptably and adequately secure system given the stakeholder's mission, capability, performance needs and concerns; the constraints imposed by stakeholder concerns related to cost, schedule, performance, risk, and loss tolerance; and other constraints associated with life cycle concepts for the system. The problem context enables the engineering team to focus on acquiring as complete an understanding of the stakeholder problem as practical, exploring all feasible solution class options, and selecting the solution class option or options to be pursued. The problem context includes:

- Determining life cycle security concepts[53]
- Defining security objectives
- Defining security requirements
- Determining measures of success

The security objectives are foundational in that they establish and scope what it means to be *adequately secure* in terms of protection against asset loss and the consequences of such loss.

[53] The term *life cycle security concept* refers to the processes and activities associated with the system throughout the life cycle (from concept development through retirement) with specific security considerations. It is an extension of the *concept of operation* and includes the processes and activities related to development, prototyping, assessment of alternative solutions, training, logistics, maintenance, sustainment, evolution, modernization, disposal, and refurbishment. Each life cycle concept has one or more security considerations and constraints that must be fully integrated into the life cycle to ensure that the system security objectives can be met. Life cycle security concepts include those applied during acquisition and program management. Life cycle security concepts can affect such things as Requests for Information, Requests for Proposal, Statements of Work, source selections, development and test environments, operating environments, supply chains, supporting infrastructures, distribution, logistics, maintenance, training, clearances, and background checks.

The security objectives have associated measures of success. The measures of success constitute specific and measurable criteria relative to operational performance measures and stakeholder concerns. Measures of success include both strength of protection and level of assurance or confidence in the protection capability that has been engineered. These measures influence the development of security requirements and assurance claims.

Life cycle security concepts are the processes, methods, and procedures associated with the system throughout its life cycle and provide distinct contexts for interpretation of system security. These concepts also serve to scope and bound attention in addressing protection needs and for broader security-informing considerations and constraints. Protection needs are determined based on the security objectives, life cycle concepts, and stakeholder concerns. The protection needs are subsequently transformed into stakeholder security requirements and associated constraints, and the measures needed to validate that all requirements have been met. A well-defined and stakeholder-validated problem definition and context provides the foundation for all systems engineering and systems security engineering and supporting activities.

The problem context may be interpreted within a life cycle phase as being informed by solutions from earlier life cycle stages, thereby providing a more accurate statement of the problem and its associated constraints. For example, the stakeholder requirements may be the “solution” of an early life cycle phase which then constrains activities completed in later life cycle stages.

2.5.2 The Solution Context

The *solution context* transforms stakeholder security requirements into derived requirements for the system, subsystem, or system element, as applicable. It also addresses the security architecture, design, and related aspects necessary to realize a system that satisfies those requirements and, lastly, produces sufficient evidence to demonstrate that those requirements have been satisfied.[54] The solution context is based on a balanced proactive and reactive system security protection strategy[55] that exercises control over events, conditions, asset loss, and the consequence of loss to the degree possible, practicable, and acceptable to stakeholders. The solution context includes:

- Defining the security aspects of the solution
- Realizing the security aspects of the solution
- Producing evidence for the security aspects of the solution

The security aspects of the solution include the development of a system protection strategy; allocated and derived security requirements; security architecture views and viewpoints; security design; security aspects, capabilities, and limitations in the system life cycle procedures; and security performance verification measures. The security aspects of the solution are realized during the implementation of the system design in accordance with the system architecture and

[54] Security constraints are transformed and incorporated into system design requirements with metadata-tagging to identify security relevance.

[55] The system security protection strategy is consistent with the overall *concept of secure function*. The concept of secure function, defined during the problem context, constitutes a strategy for a proactive and reactive protection capability throughout the system life cycle (Section D.2). The strategy has the objective to provide freedom from specific concerns associated with asset loss and loss consequences.

in satisfaction of the security requirements. The evidence associated with the security aspects of the solution is obtained with a fidelity and rigor influenced by the level of assurance[56] targeted by the security objectives. Assurance evidence is obtained from standard systems engineering verification methods (e.g., analysis, demonstration, inspection, testing, and evaluation) and complementary validation methods applied against the stakeholder requirements. Application of the solution context may be interpreted to provide a part of the solution, constraining the next iteration of the problem context.

2.5.3 The Trustworthiness Context

The *trustworthiness context* is a decision-making context that provides an evidence-based demonstration, through reasoning, that the system of interest is deemed trustworthy based on a set of claims derived from security objectives. The trustworthiness context consists of:

- Developing and maintaining the assurance case
- Demonstrating that the assurance case is satisfied

The trustworthiness context is grounded in the concept of an *assurance case*. An assurance case is a well-defined and structured set of arguments and a *body of evidence* showing that a system satisfies specific claims.[57] Assurance cases provide reasoned, auditable artifacts that support the contention that a top-level claim or set of claims is satisfied, including systematic argumentation and underlying evidence and explicit assumptions that support the claims [ISO 15026-2]. The claims may build from subclaims. For a given life cycle stage, one outcome may sufficiently satisfy a subclaim or set of subclaims, such as a subclaim that stakeholder requirements are sufficiently comprehensive to support an overall claim that the realized system is adequately secure.

An assurance case is used to demonstrate that a system exhibits some complex emergent property, such as safety, security, resilience, reliability, or survivability. An effective security assurance case contains foundational security claims that are derived from stakeholder security objectives, credible and relevant evidence that substantiates the claims, and valid arguments that relate the various evidence to the supported security claims. The result provides a compelling statement that adequate security has been achieved and driven by stakeholder needs and expectations.

Assurance cases typically include supporting information, such as assumptions, constraints, and any inferences that can affect the reasoning process. Subsequent to the development of the assurance case, analyses by subject-matter experts determine that all security claims are substantiated by the evidence produced and the arguments that relate the evidence to the claims. For maximum effectiveness, the assurance cases must be maintained in response to variances throughout the engineering effort.

The specific form of an assurance case and the level of rigor and formality in acquiring the evidence required by the assurance case is a trade space consideration. It involves the target (desired) level of assurance, the nature of the consequences for which assurance is sought, and

[56] *Assurance* is the measure of confidence associated with a given requirement. As the level of assurance increases, so does the scope, depth, and rigor associated with the methods and analyses conducted (Appendix F).

[57] Software Engineering Institute, Carnegie Mellon University.

the size and complexity of the dimensions that factor into the determination of trustworthiness. The assurance case is an *engineering construct* and must be managed accordingly to ensure that the effort expended to produce the evidence is justified by the need for that evidence in making the trustworthiness determination. The assurance claims are the key trustworthiness factor and are developed from the security objectives and associated measures of success, independent of the realization of the system and its supporting evidence.

SYSTEMS SECURITY ENGINEERING FRAMEWORK – WHY IT MATTERS

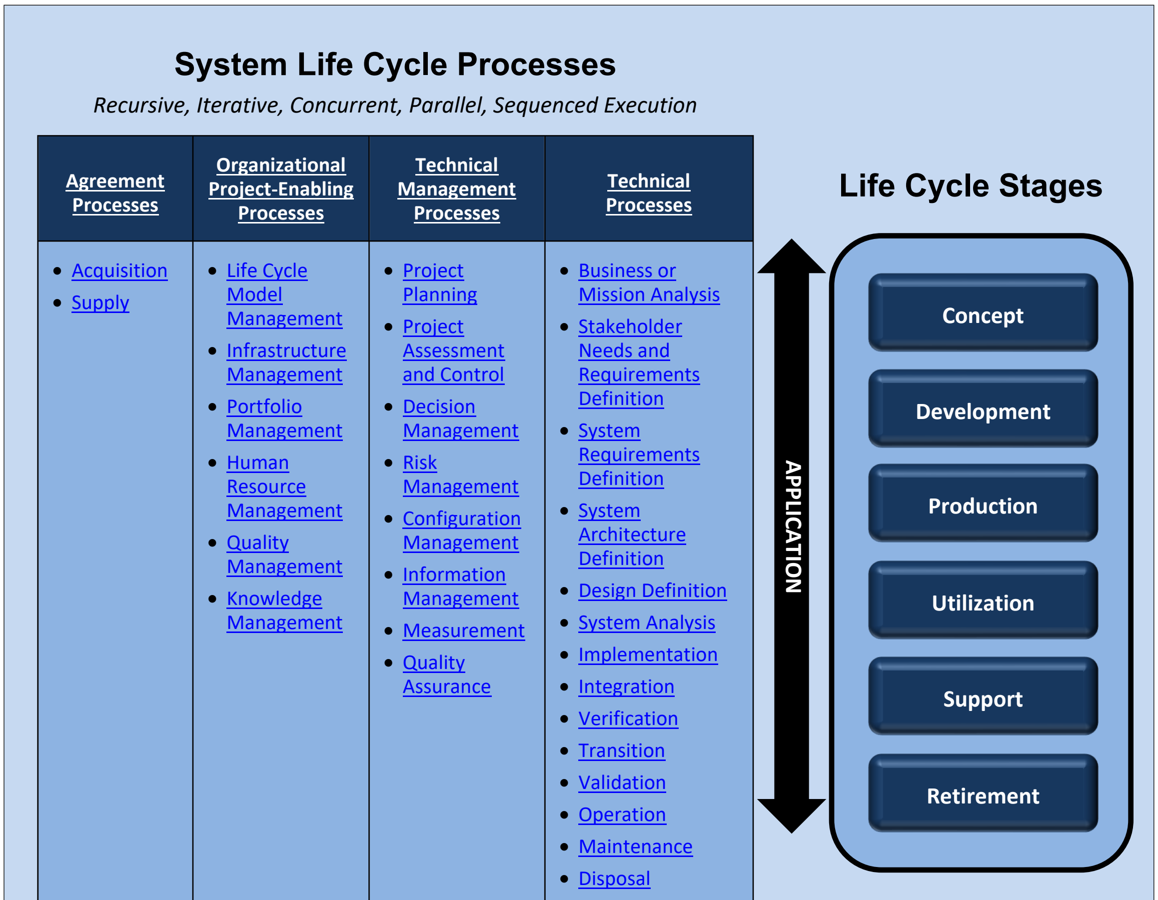
Establishing the problem, solution, and trustworthiness contexts as key components of a systems security engineering framework helps ensure that the ***security*** of a system is based on achieving a sufficiently complete understanding of the problem as defined by a set of stakeholder security objectives, security concerns, protection needs, and security requirements. This understanding is essential in order to develop effective security solutions – that is, a system that is sufficiently trustworthy and adequately secure to protect stakeholder's assets in terms of loss and the associated consequences.

CHAPTER THREE

SYSTEM LIFE CYCLE PROCESSES

SYSTEMS SECURITY IN SYSTEM LIFE CYCLE PROCESSES

This chapter describes the considerations and contributions to the system life cycle processes in [ISO 15288] to produce the behaviors and outcomes that are necessary to achieve trustworthy secure systems. The system life cycle processes are grouped into the following families: *Agreement Processes*, *Organizational Project-Enabling Processes*, *Technical Management Processes*, and *Technical Processes*. Figure 10 lists the processes and illustrates their application across the stages of the system life cycle.



Source: [ISO 15288].

FIGURE 10: SYSTEM LIFE CYCLE PROCESSES AND LIFE CYCLE STAGES

The security-related considerations and contributions to the system life cycle are provided as systems security engineering *tasks*. The tasks are aligned with the engineering viewpoints of the life cycle processes and are based on the foundational security and trust principles and concepts described in Chapter Two, Appendix C, Appendix D, Appendix E, and Appendix F. The tasks use and leverage the principles, concepts, terms, and practices of systems engineering to facilitate consistency in their application as part of a systems engineering effort.

This publication is not intended to be prescriptive in nature. The processes, activities, and tasks are to be applied as needed. They are not dependent on, oriented to, or presumed to be used in any particular system development methodology. By design, the processes, activities, and tasks can be applied concurrently, iteratively, or recursively: (1) at any level in the structural hierarchy of a system, (2) with the appropriate fidelity and rigor, and (3) at any stage in the system life cycle in accordance with acquisition, systems engineering, or other process models.[58] The application of the processes, activities, or tasks relies on the skill, expertise, and experience of the *practitioner*.

The system life cycle processes are intended to be tailored to achieve optimized and efficient results.[59] Tailoring can include:

- Applying the system life cycle processes to an organization's preferred development model
- Ordering or sequencing the activities and tasks in the system life cycle processes
- Accomplishing the outcomes in ways that do not strictly adhere to the presentation of the processes in this publication
- Supplementing the activities and tasks to achieve specific outcomes

Tailoring may be motivated by the stage of the system life cycle; the size, scope, and complexity of the system; specialized requirements; or the need to accommodate specific technologies, methods, or techniques used to develop the system. Tailoring may be appropriate when the activities of different processes overlap or interact in ways not defined in this publication.[60] Tailoring the system life cycle processes allows the engineering team to:

- Optimize the application of the processes in response to technological, programmatic, acquisition, process, procedural, system life cycle stage, or other objectives and constraints
- Allow for concurrent application of the processes by sub-teams focused on different parts of the same engineering effort
- Facilitate the application of the processes to conform with a variety of system development methodologies, processes, and models (e.g., agile, spiral, waterfall) that could be used on a single engineering effort

[58] Systems engineering and system life cycle processes do not map explicitly to specific stages in the system life cycle. Rather, the processes may occur in one or more stages of the life cycle depending on the particular process and the conditions associated with the systems engineering effort. For example, the *Maintenance* process includes activities that plan the maintenance strategy such that it is possible to identify constraints on the system design necessitated by how the maintenance will be performed once the system is operational. Therefore, the *Maintenance Process* is conducted prior to or concurrent with the *Design Definition* process.

[59] Tailoring can occur as part of the project planning process at the start of the systems-engineering effort or in an ad hoc manner at any time during the engineering effort when situations and circumstances so dictate. Understanding the fundamentals of systems security engineering (i.e., the science underpinning the discipline) helps to inform the tailoring process whenever it occurs during the system life cycle. The INCOSE Systems Engineering Handbook provides additional guidance on how to tailor the systems engineering processes [INCOSE14].

[60] For example, the engineering team may need to initiate a system modification in a relatively short period to respond to a serious security incident. In this situation, the team may only informally consider each process rather than formally executing each process. It is essential that any system modifications continue to support stakeholder protection needs. Without a system-level perspective, modifications could fix one problem while introducing others.

- Accommodate the need for unanticipated or other event-driven execution of processes to resolve issues and respond to changes that occur during the engineering effort

While the life cycle processes from [ISO 15288] are addressed in terms of systems security engineering, the activities and tasks in this publication are neither a restatement of those processes nor do they constitute a one-for-one mapping to those processes. This publication focuses on specific contributions to the process, and the activities and tasks are titled to reflect the security contributions. In some cases, activities and tasks have been added to address the range of outcomes appropriate for the achievement of trustworthy, secure system objectives.

The descriptions of the system life cycle processes assume that sufficient time, funding, and human and material resources are available to ensure a complete application of the processes within the systems engineering effort. The life cycle processes represent the "standard of excellence" within which appropriate tailoring is accomplished to achieve realistic, optimal, and cost-effective results within the constraints imposed on the engineering team.

Each of the system life cycle processes contains a set of *activities* and *tasks* that produce a set of security-focused *outcomes*.[61] These outcomes combine to deliver a system and corresponding body of evidence that serve as the basis to:

- Substantiate the security and the trustworthiness of the system
- Determine security risk across stakeholder concerns and with respect to the use of the system in support of mission or business objectives
- Help stakeholders decide which operational constraints are necessary to mitigate security risk
- Provide inputs to other processes associated with delivering the system
- Support the system throughout the stages of its life cycle[62]

Each system life cycle process description has the following sections:

- **Life Cycle Purpose:** Describes the goals of performing the process [ISO 15288].
- **Security Purpose:** Establishes what the process achieves from the security standpoint.
- **Security Outcomes:** Expresses the security-related observable results expected from the successful performance of the process and the data generated by the process.[63]

[61] Outcomes from the systems engineering processes inform other systems engineering processes and can also serve to inform processes external to the engineering effort, such as the organizational life cycle processes of stakeholders and certification, authorization, or regulatory processes.

[62] The comprehensiveness, depth, fidelity, credibility, and relevance of the body of evidence are factors in helping to achieve the level of assurance sought by stakeholders. The objective is to have a body of evidence that is sufficient to convince stakeholders that their assurance needs are satisfied. The assurance level is an engineering trade space factor that must be planned for and executed with the appropriate fidelity and rigor. Assurance considerations can affect system cost and development schedule.

[63] The data and information generated during the execution of a process is not necessarily produced in the form of a document. Such data and information can be conveyed in the most effective manner as set forth by stakeholders or the engineering team. Data and information produced during a particular process may flow into a subsequent process or support other processes that are associated with the systems security engineering process.

- **Security Activities:** Provides a set of cohesive security-related tasks that support achievement of the security outcomes for the process. The tasks are accomplished cooperatively within and across various roles of the organization, inclusive of systems security engineering. While this publication focuses on the scope and responsibility of systems security engineering, it is not the case that all aspects of every task are fulfilled by systems security engineering.

The following naming convention is established for the system life cycle processes. Each process is identified by a two-character designation (e.g., BA is the official designation for the *Business or Mission Analysis* process). Table 3 lists the system life cycle processes and their associated two-character designators.

TABLE 3: PROCESS NAMES AND DESIGNATORS

ID	PROCESS	ID	PROCESS
AQ	Acquisition	MS	Measurement
AR	System Architecture Definition	OP	Operation
BA	Business or Mission Analysis	PA	Project Assessment and Control
CM	Configuration Management	PL	Project Planning
DE	Design Definition	PM	Portfolio Management
DM	Decision Management	QA	Quality Assurance
DS	Disposal	QM	Quality Management
HR	Human Resource Management	RM	Risk Management
IF	Infrastructure Management	SA	System Analysis
IM	Information Management	SN	Stakeholder Needs and Requirements Definition
IN	Integration	SP	Supply
IP	Implementation	SR	System Requirements Definition
KM	Knowledge Management	TR	Transition
LM	Life Cycle Model Management	VA	Validation
MA	Maintenance	VE	Verification

The security activities and tasks in each system life cycle process are uniquely identified using a two-character designation plus a numerical designation. For example, the first activity in the *Stakeholder Needs and Requirements Definition* process is designated SN-1. The first two tasks within SN-1 are designated SN-1.1 and SN-1.2, respectively. The identification of the activities and tasks within each system life cycle process provides for precise referencing and traceability among the process elements. Task descriptions may contain a *notes* section that provides additional information on considerations relevant to the successful execution of that task. A *references* section provides a list of pertinent publications related to the activity and is a source of content for additional information. Finally, a *related publications* section provides a list of documents that are related to the topic being addressed in the activity. The remaining sections in this chapter describe the security contributions, considerations, and outcomes for the 30 system life cycle processes defined in [ISO 15288].

Finally, the outcomes described in this publication are achieved by personnel and machines. Personnel conduct activities and tasks, such as those defined in the [ISO 15288] system life cycle processes, to produce outcomes that achieve the defined security objectives. There is no single personnel role that is responsible to produce all of the outcomes stated in the system life cycle processes (i.e., the life cycle processes are not role-specific). Thus, there may be multiple roles that contribute to a specific outcome.

This publication describes the engineering *considerations*, not the engineering responsibilities, to produce the specified outcomes. Those responsibilities reside with the organizations using the guidance in this publication. This facilitates maximum flexibility for organizations to define, combine, and allocate responsibility to support the execution of the life cycle processes. There is no expectation that any particular role or title is assigned any specific responsibility or possesses any specific authority. Figure 11 provides an example of the types of personnel and roles that support the system life cycle processes. Each personnel category has a scope of authority, control, and responsibility and a variety of roles that collectively achieve the outcomes for the category. Collectively, the outcomes produced across all categories achieve the defined security objectives.

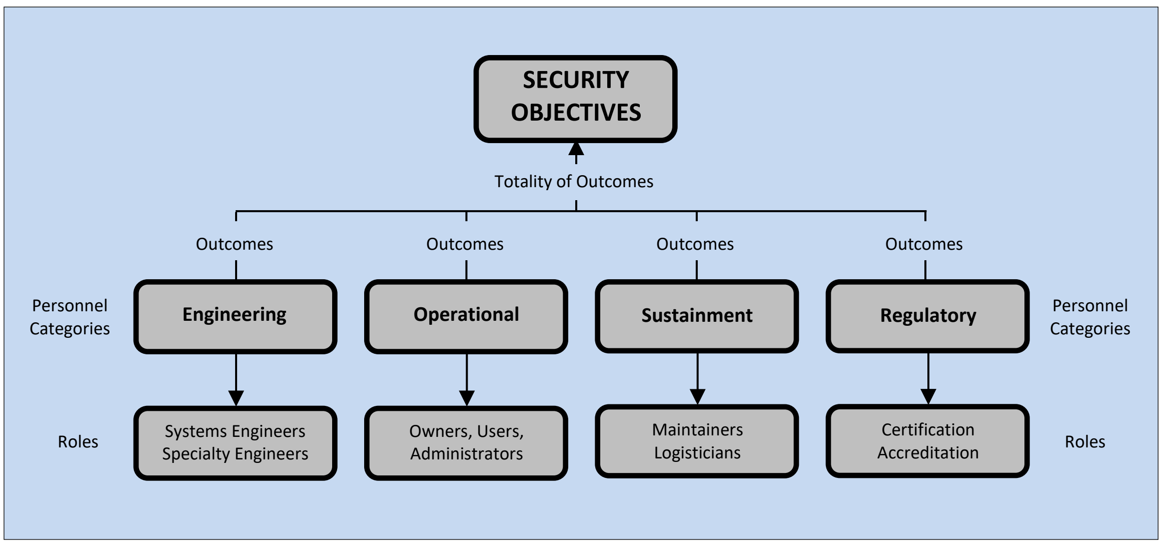


FIGURE 11: TYPES OF PERSONNEL AND ROLES THAT SUPPORT LIFE CYCLE PROCESSES

3.1 AGREEMENT PROCESSES

This section contains the *Agreement Processes* from [ISO 15288] with security-related considerations and contributions.

3.1.1 Acquisition

The purpose of the *Acquisition* process is to obtain a product or service in accordance with the acquirer's requirements.

[**ISO 15288**] *Reprinted with permission from IEEE, Copyright IEEE 2015, All rights reserved.*

3.1.1.1 Security Purpose

- To obtain a product or service in accordance with the acquirer's security requirements

3.1.1.2 Security Outcomes

- A request for supply includes security criteria.

- One or more suppliers are selected that satisfy the security criteria.
- An agreement containing security criteria is established between the acquirer and the supplier.
- A product or service complying with the security criteria in the agreement is accepted.
- The security aspects of acquirer obligations defined in the agreement are satisfied.

3.1.1.3 Security Activities and Tasks

AQ-1 PREPARE FOR THE ACQUISITION

AQ-1.1 Define the security aspects of the strategy for how the acquisition will be conducted.

Note: This strategy describes or references the life cycle model, security risks and issues mitigation, a schedule of security-relevant milestones, protection of acquirer and supplier assets, and security-relevant selection criteria if the supplier is external to the acquiring organization. It also includes key security drivers and security-relevant characteristics of the acquisition, such as responsibilities and liabilities; specific models, methods, or processes; formality; level of criticality; and security's priority within relevant trade-off factors.

AQ-1.2 Prepare a request for a product or service that includes the security requirements.

Note: The request includes security criteria for the business practices with which the supplier is to comply, a list of bidders with adequate security qualifications, and the security criteria that will be used to select the supplier.

References: [ISO 15288, Section 6.1.1.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.1.3.1]; [ISO 21827].

AQ-2 ADVERTISE THE ACQUISITION AND SELECT THE SUPPLIER

AQ-2.1 Securely communicate the request for the supply of a product or service to potential suppliers.

AQ-2.2 Select one or more suppliers that meet the security criteria.

References: [ISO 15288, Section 6.1.1.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Sections 6.1.1.3.2, 6.1.1.3.3]; [ISO 21827].

AQ-3 ESTABLISH AND MAINTAIN AN AGREEMENT

AQ-3.1 Develop and approve an agreement with the supplier that includes security acceptance criteria.

Note: This agreement ranges in formality from a written contract to a verbal agreement. Appropriate to the level of formality, the agreement establishes security requirements, secure development and delivery milestones, security verification, security validation and security aspects of acceptance conditions, security aspects of process requirements (e.g., configuration management, risk management, and measurement), and security aspects of handling of data rights and intellectual property so that both parties of the agreement understand the basis for executing the agreement. The security aspects of the agreement also include application of all of the above to subcontractors and other supporting organizations to the supplier.

AQ-3.2 Identify necessary security-relevant changes to the agreement.

AQ-3.3 Evaluate the security impact of changes to the agreement.

Note: The basis for the agreement change may or may not be security related. However, there may be security-related impact regardless of the basis for the change.

AQ-3.4 Update the security criteria in the agreement with the supplier, as necessary.

References: [ISO 15288, Section 6.1.1.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.1.3.4]; [ISO 21827].

AQ-4 MONITOR THE AGREEMENTS

AQ-4.1 Assess the execution of the security aspects of the agreement.

Note: This includes confirmation that all parties are meeting their security-relevant responsibilities according to the agreement.

AQ-4.2 Securely provide data needed by the supplier, and resolve issues in a timely manner.

References: [ISO 15288, Section 6.1.1.3 d)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.1.3.5]; [ISO 21827].

AQ-5 ACCEPT THE PRODUCT OR SERVICE

AQ-5.1 Confirm that the delivered product or service complies with the security aspects of the agreement.

AQ-5.2 Securely provide payment or other agreed consideration.

AQ-5.3 Accept the product or service from the supplier or other party, as directed by the security criteria in the agreement.

AQ-5.4 Close the agreement in accordance with agreement security criteria.

References: [ISO 15288, Section 6.1.1.3 e)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.1.3.6]; [ISO 21827].

3.1.2 Supply

The purpose of the *Supply* process is to provide an acquirer with a product or service that meets agreed requirements.

3.1.2.1 Security Purpose

- To provide an acquirer with a product or service that meets agreed security requirements

3.1.2.2 Security Outcomes

- A response to the acquirer's request addresses the acquirer's security requirements.
- An agreement established between the acquirer and supplier includes security requirements.
- A product or service that satisfies the acquirer's security requirements is provided.
- Supplier security obligations defined in the agreement are satisfied.

- Responsibility for the acquired product or service, as directed by the agreement, is securely transferred.

3.1.2.3 Security Activities and Tasks

SP-1 PREPARE FOR THE SUPPLY

SP-1.1 Identify the security aspects of an acquirer's need for a product or service.

SP-1.2 Define the security aspects of the supply strategy.

Note: This strategy describes or references the security aspects of the life cycle model, risks and issues mitigation, and a schedule of security-relevant milestones. It also includes key security-relevant drivers and characteristics of the acquisition such as responsibilities and liabilities, specific security-related models, security-relevant methods or processes, level of criticality, formality, and priority of relevant trade-off factors.

References: [ISO 15288, Section 6.1.2.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.2.3.1]; [ISO 21827].

SP-2 RESPOND TO A REQUEST FOR SUPPLY OF PRODUCTS OR SERVICES

SP-2.1 Evaluate a request for a product or service to determine the security-relevant feasibility and how to respond.

SP-2.2 Prepare a response that satisfies the security criteria in the solicitation.

References: [ISO 15288, Section 6.1.2.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.2.3.2]; [ISO 21827].

SP-3 ESTABLISH AND MAINTAIN AN AGREEMENT

SP-3.1 Negotiate and approve an agreement with the acquirer that includes security acceptance criteria.

Note 1: This includes configuration management, risk reporting, reporting of security measures, and security measure analysis; security requirements; secure development; security verification; security validation; security acceptance procedures and criteria; regulatory body acceptance, authorization, and approval; procedures for transport, handling, delivery, and storage; security and privacy protections and restrictions on the use, dissemination, and destruction of data, information, and intellectual property; security-relevant exception-handling procedures and criteria; agreement change management procedures; and agreement termination procedures.

Note 2: The security aspects of the agreement also include the application of all of the above to the plans for use of subcontractors.

SP-3.2 Identify necessary security-relevant changes to the agreement.

SP-3.3 Evaluate the security impact of necessary changes to the agreement.

Note: The basis for the agreement change may or may not be security related. However, there may be security-related impact regardless of the basis for the change. A security-related evaluation of the needed change identifies any security relevance and determines impact in terms of plans, schedule, cost, technical capability, quality, assurance, and trustworthiness.

SP-3.4 Update the security criteria in the agreement with the acquirer, as necessary.

References: [ISO 15288, Section 6.1.2.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.2.3.3]; [ISO 21827].

SP-4 EXECUTE THE AGREEMENT

SP-4.1 Execute the security aspects of the agreement according to established project plans.

Note: A suppler sometimes adopts or agrees to use acquirer processes, including security-relevant processes.

SP-4.2 Assess the execution of the security aspects of the agreement.

Note: This includes confirmation that all parties are meeting their security responsibilities according to the agreement.

References: [ISO 15288, Section 6.1.2.3 d)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.2.3.4]; [ISO 21827].

SP-5 DELIVER AND SUPPORT THE PRODUCT OR SERVICE

SP-5.1 Deliver the product or service in accordance with the agreement security criteria.

SP-5.2 Provide security assistance to the acquirer, per the agreement.

SP-5.3 Securely accept and acknowledge payment or other agreed consideration.

SP-5.4 Transfer the product or service to the acquirer or other party as directed by the security requirements in the agreement.

Note: This includes the transfer of hardware, software, and sensitive, proprietary, and classified information.

SP-5.5 Close the agreement in accordance with the agreement security criteria.

References: [ISO 15288, Section 6.1.2.3 e)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.1.2.3.5]; [ISO 21827].

3.2 ORGANIZATIONAL PROJECT-ENABLING PROCESSES

This section contains the *Organizational Project-Enabling Processes* from [ISO 15288] with security-related considerations and contributions.

3.2.1 Life Cycle Model Management

The purpose of the *Life Cycle Model Management* process is to define, maintain, and help ensure the availability of policies, life cycle processes, life cycle models, and procedures for use by the organization with respect to the scope of this International Standard.

3.2.1.1 Security Purpose

- To help ensure that security needs and considerations are incorporated in policies, life cycle processes, life cycle models, and procedures used by the organization

3.2.1.2 Security Outcomes

- Security considerations are captured in organizational policies and procedures for the management and deployment of life cycle models and processes.
- Security roles, responsibility, accountability, and authority within life cycle policies, processes, models, and procedures are defined.
- The selection of policies, life cycle processes, life cycle models, and procedures for use by the organization is informed by security needs and considerations.
- Security needs and considerations for policies, life cycle processes, life cycle models, and procedures for use by the organization are assessed.
- Prioritized security-relevant process, model, and procedure improvements are implemented.

3.2.1.3 Security Activities and Tasks

LM-1 ESTABLISH THE LIFE CYCLE PROCESSES

LM-1.1 Establish policies and procedures for process management and deployment that are consistent with the security aspects of organizational strategies.

Note: The policies and procedures may be security focused, security based, or may have security-informing aspects.

LM-1.2 Establish the security aspects of the life cycle processes that implement the requirements of [ISO 15288] and that are consistent with organizational strategies.

LM-1.3 Define the security roles, responsibilities, accountabilities, and authorities to facilitate implementation of the security aspects of life cycle processes and the strategic management of life cycles.

LM-1.4 Define the security aspects of the criteria that control progression through the life cycle.

Note: This includes security criteria for gates, checkpoints, and entry/exit criteria for milestones and decision points.

LM-1.5 Establish security criteria for the standard life cycle models for the organization, including criteria for outcomes for each stage.

Note: The life cycle model comprises one or more stages, as needed, with each stage having security aspects to its purpose and outcomes. The model is assembled as a sequence of stages that overlap or iterate as appropriate for the scope of the system of interest, magnitude, complexity, changing needs, and opportunities (including protection needs and opportunities). The life cycle processes and activities are selected, tailored as appropriate, and employed in a stage to fulfill the security aspects of the purpose and outcomes of that stage.

References: [ISO 15288, Section 6.2.1.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.2.1.3.1]; [ISO 21827]; [DoDD 8140.01].

LM-2 ASSESS THE LIFE CYCLE PROCESS

LM-2.1 Monitor the security aspects of process execution across the organization.

Note: This includes the analysis of process measures and the review of security-relevant trends with respect to strategic security criteria, feedback from the projects regarding the effectiveness and efficiency of the processes, and monitoring execution according to regulations and organizational policies.

LM-2.2 Conduct reviews of the security aspects of the life cycle models used by the projects.

Note: This includes confirming the suitability, adequacy, and effectiveness of the life cycle models used by the project. The reviews should be conducted periodically and be event-driven, (e.g., at completions of large project milestones).

LM-2.3 Identify security-relevant improvement opportunities from assessment results.

References: [ISO 15288, Section 6.2.1.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.2.1.3.2]; [ISO 21827].

LM-3 IMPROVE THE PROCESS

LM-3.1 Prioritize and plan for security-relevant improvement opportunities.

LM-3.2 Implement security improvement opportunities, and inform relevant stakeholders.

Note: This includes regulatory, certification, accreditation, acceptance, and similar stakeholders.

References: [ISO 15288]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.2.1.3.3]; [ISO 21827].

3.2.2 Infrastructure Management

The purpose of the *Infrastructure Management* process is to provide infrastructure and services to projects to support organization and project objectives throughout the life cycle.

3.2.2.1 Security Purpose

- To define protection needs for the aspects of infrastructure and services that support organization and project objectives

3.2.2.2 Security Outcomes

- Protection needs for the infrastructure are defined.
- Security capabilities and constraints of infrastructure elements are specified.
- Infrastructure elements that satisfy infrastructure security specifications are obtained.
- Secure infrastructure is available.
- Prioritized infrastructure security-relevant improvements are implemented.

3.2.2.3 Security Activities and Tasks

IF-1 ESTABLISH THE INFRASTRUCTURE

IF-1.1 Define the infrastructure security protection needs.

Note: The security aspects of infrastructure resource needs are considered in context with other projects and resources within the organization. Security constraints that influence and control the provision of infrastructure resources and services for the project are also defined.

IF-1.2 Identify, obtain, and provide the infrastructure resources and services that satisfy the security protection needs to securely implement and support projects.

References: [ISO 15288, Section 6.2.2.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Sections 6.2.2.3.1, 6.2.2.3.2]; [ISO 21827].

IF-2 MAINTAIN THE INFRASTRUCTURE

IF-2.1 Evaluate the degree to which delivered infrastructure resources satisfy project protection needs.

IF-2.2 Identify and provide security improvements or changes to infrastructure resources as project requirements change.

Note: Any mismatch between project security needs and the security provided by infrastructure resources may result in gaps in assurance.

References: [ISO 15288, Section 6.2.2.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.2.2.3.3]; [ISO 21827].

3.2.3 Portfolio Management

The purpose of the *Portfolio Management* process is to initiate and sustain necessary, sufficient, and suitable projects in order to meet the strategic objectives of the organization.

3.2.3.1 Security Purpose

- To identify security considerations for the projects that meet the strategic objectives of the organization

3.2.3.2 Security Outcomes

- Security aspects of strategic venture opportunities, investments, or necessities are prioritized.
- Security aspects of projects are identified.
- Resources and budgets for the security aspects of each project are allocated.
- Project management responsibilities, accountability, and authorities for security are defined.
- Projects that meet the security criteria in agreements and stakeholder security requirements are sustained.
- Projects that do not meet the security criteria in agreements or do not satisfy stakeholder security requirements are redirected or terminated.
- Projects that have completed the security aspects of agreements and that satisfy all stakeholder security requirements are closed.

3.2.3.3 Security Activities and Tasks

PM-1 DEFINE AND AUTHORIZE PROJECTS

PM-1.1 Identify potential new or modified security capabilities or missions.

Note: The organization strategy, concept of operations, or gap or opportunity analysis is reviewed to identify security-driven gaps, problems, or opportunities.

PM-1.2 Identify security aspects of potential new or modified capabilities or missions.

Note: The organization strategy, concept of operations, or gap or opportunity analysis is reviewed to identify security-relevant gaps, problems, or opportunities.

PM-1.3 Prioritize, select, and establish new business opportunities, ventures, or undertakings with consideration for security objectives and concerns.

PM-1.4 Define the security aspects of projects, accountabilities, and authorities.

Note: This includes project proprietary, sensitivity, and privacy criteria.

PM-1.5 Identify the security aspects of expected goals, objectives, and outcomes of each project.

Note: This includes project proprietary, sensitivity, and privacy criteria.

PM-1.6 Identify and allocate resources for the achievement of the security aspects of project goals and objectives.

PM-1.7 Identify the security aspects of any multi-project interfaces and dependencies to be managed or supported by each project.

Note: This includes interfaces and dependencies with enabling systems and services, as well as all associated data and information.

PM-1.8 Specify the security aspects of project reporting requirements, and review milestones that govern the execution of each project.

PM-1.9 Authorize each project to commence execution of project plans, including its security aspects.

References: [ISO 15288], Section 6.2.3.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207], Section 6.2.3.3.1]; [ISO 21827].

PM-2 EVALUATE THE PORTFOLIO OF PROJECTS

PM-2.1 Evaluate the security aspects of projects to confirm ongoing viability.

Note: This includes the following:

- The project is making progress towards achieving established security goals and objectives.
- The project is complying with project security directives.
- The project is being conducted according to security aspects of project life cycle policies, processes, and procedures.
- The project remains viable, as indicated by the continuing need for security services, practical secure product implementation, and acceptable security-driven investment benefits.

PM-2.2 Act to continue projects that are satisfactorily progressing in consideration of project security aspects.

PM-2.3 Act to redirect projects that can be expected to progress satisfactorily with appropriate security-informed redirection.

References: [ISO 15288], Section 6.2.3.3 b)].

Related Publications: [ISO 12207, Section 6.2.3.3.2]; [ISO 21827].

PM-3 TERMINATE PROJECTS

PM-3.1 Where agreements permit, act to cancel or suspend projects whose security-driven disadvantages or security-driven risks to the organization outweigh the benefits of continued investments.

PM-3.2 After completion of the agreement for the security aspects of products or services, act to close the projects.

Note: Closure is accomplished in accordance with organizational security policies, procedures, and the agreement.

References: [ISO 15288, Section 6.2.3.3 c)].

Related Publications: [ISO 12207], Section 6.2.3.3.3]; [ISO 21827].

3.2.4 Human Resource Management

The purpose of the *Human Resource Management* process is to provide the organization with necessary human resources and to maintain their competencies in a manner consistent with strategic needs.

3.2.4.1 Security Purpose

- To define the security criteria for necessary human resources and maintain their competencies in a manner consistent with strategic needs

3.2.4.2 Security Outcomes

- Security-relevant skills required by projects are identified.
- Personnel with necessary security skills are provided to projects.
- Security-relevant skills of personnel are developed, maintained, or enhanced.
- Security-relevant personnel conflicts are resolved.

3.2.4.3 Security Activities and Tasks

HR-1 IDENTIFY SKILLS

HR-1.1 Identify the security-relevant skills needed based on current and expected projects.

HR-1.2 Identify and record security-relevant skills of personnel.

References: [ISO 15288, Section 6.2.4.3 a)].

Related Publications: [ISO 12207, Section 6.2.4.3.1]; [ISO 21827]; [ISO 27034-1]; [SP 800-181] [DoDD 8140.01].

HR-2 DEVELOP SKILLS

HR-2.1 Establish a plan for security-relevant skills development.

Note: The security-relevant skills include core and specialty competencies.

HR-2.2 Obtain security-relevant training, education, or mentoring resources.

HR-2.3 Provide planned security-relevant skills development.

HR-2.4 Maintain records of security-relevant skills development.

References: [ISO 15288, Section 6.2.4.3 b)].

Related Publications: [ISO 12207, Section 6.2.4.3.2]; [ISO 21827]; [ISO 27034-1]; [DoDD 8140.01].

HR-3 ACQUIRE AND PROVIDE SKILLS

HR-3.1 Obtain qualified personnel when security-relevant skill deficits are identified.

HR-3.2 Maintain and manage the pool of security-skilled personnel necessary to staff ongoing projects.

HR-3.3 Make personnel assignments based on security-relevant project and staff development needs.

HR-3.4 Motivate security-skilled personnel (e.g., through career development and reward mechanisms).

HR-3.5 Resolve the security aspects of personnel conflicts across or within projects.

Note: Conflicts across or within projects may include personnel capacity, availability, qualification conflicts, and personality conflicts.

References: [ISO 15288] 15288, Section 6.2.4.3 c).

Related Publications: [ISO 12207, Section 6.2.4.3.3; [SP 800-181].

3.2.5 Quality Management

The purpose of the *Quality Management* process is to assure that products, services, and implementations of the quality management process meet organizational and project quality objectives and achieve customer satisfaction.

3.2.5.1 Security Purpose

- To define organizational and project security quality objectives and the criteria used to determine that products, services, and implementations of the *Quality Management* process meet those security objectives

3.2.5.2 Security Outcomes

- Organizational security quality management policies, standards, and procedures are defined and implemented.
- Security quality evaluation criteria and methods are established.
- Resources and information are provided to projects to support the operation and monitoring of project security quality assurance activities.
- Security aspects of quality evaluation results are analyzed.

- Security quality management policies and procedures are improved based on project and organization results.

3.2.5.3 Security Activities and Tasks

QM-1 PLAN QUALITY MANAGEMENT

QM-1.1 Establish the security aspects of quality management policies, standards, and procedures.

QM-1.2 Define responsibilities and authority for the implementation of security quality management.

QM-1.3 Define security quality evaluation criteria and methods.

QM-1.4 Provide resources, data, and information for security quality management.

References: [ISO 15288, Section 6.2.5.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 9001].

Related Publications: [ISO 12207, Section 6.2.5.3.1].

QM-2 ASSESS QUALITY MANAGEMENT

QM-2.1 Gather and analyze quality assurance evaluation results in accordance with the defined security quality evaluation criteria.

QM-2.2 Assess customer satisfaction.

Note: The satisfaction focuses on security for the systems security efforts.

QM-2.3 Conduct periodic reviews of project quality assurance activities for compliance with the security quality management policies, standards, and procedures.

QM-2.4 Monitor the status of security quality improvements on processes, products, and services.

References: [ISO 15288, Section 6.2.5.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 9001].

Related Publications: [ISO 12207, Section 6.2.5.3.1].

QM-3 PERFORM QUALITY MANAGEMENT CORRECTIVE AND PREVENTIVE ACTIONS

QM-3.1 Plan corrective actions when security quality management objectives are not achieved.

QM-3.2 Plan preventive actions when there is a sufficient risk that security quality management objectives will not be achieved.

QM-3.3 Monitor the security aspects of corrective and preventive actions to completion and inform stakeholders.

References: [ISO 15288, Section 6.2.5.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 9001].

Related Publications: [ISO 12207], Section 6.2.5.3.2].

3.2.6 Knowledge Management

The purpose of the *Knowledge Management* process is to create the capability and assets that enable the organization to exploit opportunities to reapply existing knowledge.

3.2.6.1 Security Purpose

- To enable the organization to exploit opportunities to reapply existing security knowledge

3.2.6.2 Security Outcomes

- A taxonomy for the application of security-relevant knowledge assets is identified.
- Organizational security knowledge, skills, and knowledge assets are organized.
- Organizational security knowledge, skills, and knowledge assets are available.
- Organizational security knowledge, skills, and knowledge assets are communicated across the organization.
- Security knowledge management usage data is analyzed.

3.2.6.3 Security Activities and Tasks

KM-1 PLAN KNOWLEDGE MANAGEMENT

KM-1.1 Define the security aspects of the knowledge management strategy.

Note: The security aspects of the knowledge management strategy generally include:

- Identifying security knowledge domains and technologies and their potential for the reapplication of knowledge
- Plans for obtaining and maintaining security knowledge, skills, and security knowledge assets for their useful life
- Characterization of the types of security knowledge, security skills, and security knowledge assets to be collected and maintained
- Criteria for accepting, qualifying, and retiring security knowledge, security skills, and security knowledge assets
- Procedures for controlling changes to the security knowledge, security skills, and security knowledge assets
- Plans, mechanisms, and procedures for protection, control, and access to classified or sensitive data and information
- Mechanisms for secure storage and secure retrieval

KM-1.2 Identify the security knowledge, skills, and knowledge assets to be managed.

KM-1.3 Identify projects that can benefit from the application of the security knowledge, skills, and knowledge assets.

References: [ISO 15288, Section 6.2.6.3 a)].

Related Publications: [ISO 12207, Section 6.2.4.3.4]; [ISO 21827]; [SP 800-181]; [DoDD 8140.01].

KM-2 SHARE KNOWLEDGE AND SKILLS THROUGHOUT THE ORGANIZATION

KM-2.1 Establish and maintain a classification for capturing and sharing security knowledge and skills.

Note: This classification includes security expert, common security, and security domains knowledge and skills, as well as lessons learned.

KM-2.2 Capture or acquire security knowledge and skills.

KM-2.3 Make security knowledge and skills accessible across the organization.

References: [ISO 15288, Section 6.2.6.3 b)].

Related Publications: [ISO 12207, Section 6.2.4.3.4]; [ISO 21827].

KM-3 SHARE KNOWLEDGE ASSETS THROUGHOUT THE ORGANIZATION

KM-3.1 Establish a taxonomy to organize security knowledge assets.

Note: The taxonomy includes the following:

- Definition of the boundaries of security domains and their relationships to one another
- Definition of the boundaries of security-related domains (e.g., safety) and their relationships to one another
- Domain models that capture essential common and different security-relevant features, capabilities, concepts, and functions

KM-3.2 Develop or acquire security knowledge assets.

Note: Security knowledge assets include system elements or their representations (e.g., reusable code libraries, security reference architectures), architecture or design elements (e.g., security architecture or security design patterns), processes, security criteria, or other technical information (e.g., training materials) related to security domain knowledge and lessons learned.

KM-3.3 Make all knowledge assets securely accessible to the organization.

References: [ISO 15288, Section 6.2.6.3 c)]; [ISO 42010].

Related Publications: [ISO 12207, Section 6.2.4.3.4]; [ISO 21827].

KM-4 MANAGE KNOWLEDGE, SKILLS, AND KNOWLEDGE ASSETS

KM-4.1 Maintain security knowledge, skills, and knowledge assets.

KM-4.2 Monitor and record the use of security knowledge, skills, and knowledge assets.

KM-4.3 Periodically reassess the currency of the security aspects of technology and market needs of the security knowledge assets.

References: [ISO 15288, Section 6.2.6.3 d)].

Related Publications: [ISO 12207, Section 6.2.4.3.4]; [ISO 21827].

3.3 TECHNICAL MANAGEMENT PROCESSES

This section contains the *Technical Management Processes* from [ISO 15288] with security-related considerations and contributions.

3.3.1 Project Planning

The purpose of the *Project Planning* process is to produce and coordinate effective and workable plans.

3.3.1.1 Security Purpose

- To determine and coordinate the security aspects of effective and workable plans

3.3.1.2 Security Outcomes

- Security objectives, security-specific plans, and the security aspects of other plans are defined.
- Security-relevant roles, responsibilities, accountabilities, and authorities within the project are defined.
- Security aspects of performance and achievement criteria are defined.
- The resources and services necessary to achieve the security objectives are committed.
- Plans for the execution of the security aspects of the project are activated.

3.3.1.3 Security Activities and Tasks

PL-1 DEFINE THE PROJECT

PL-1.1 Identify the security aspects of project objectives and constraints.

Note: Objectives and constraints include strategic security, assurance, and trustworthiness goals, as well as loss thresholds and regulatory concerns. Each security-relevant objective is identified with a level of detail that permits selection, tailoring, and implementation of the appropriate processes and activities.

PL-1.2 Define the security aspects of the project scope as established in agreements.

Note: This includes the relevant activities required to satisfy security aspects of decision criteria and complete the project successfully.

PL-1.3 Define and maintain security views of the project life cycle model that are comprised of stages using the defined life cycle models of the organization.

PL-1.4 Establish appropriate security aspects of the breakdown structures.

Note: Each security-relevant element of a breakdown structure is described with a level of detail that is consistent with identified security risks and required visibility.

PL-1.5 Define and maintain the security aspects of processes that will be applied on the project.

References: [ISO 15288, Section 6.3.1.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3]; [ISO 24748-1].

Related Publications: [ISO 12207, Section 6.3.1.3.1]; [ISO 21827].

PL-2 PLAN PROJECT AND TECHNICAL MANAGEMENT

PL-2.1 Define and maintain the security aspects of a project schedule based on management and technical objectives and work estimates.

Note: This includes security aspects that impact the definition of the duration, relationship, dependencies, and sequence of activities; achievement milestones; resources employed; reviews (including security subject matter expertise employed); and schedule reserves for security risk management necessary to achieve timely completion of the project.

PL-2.2 Define the security aspects of achievement criteria for the life cycle decision gates, delivery dates, and major dependencies on external inputs and outputs.

Note: This includes criteria defined by regulatory, certification, evaluation, and other approval authorities.

PL-2.3 Define the security aspects of project performance criteria.

PL-2.4 Define the security-related project costs, and plan the budget.

PL-2.5 Define the security-relevant roles, responsibilities, accountabilities, and authorities.

Note: This includes defining the project organization, staff acquisitions, and development of staff security-relevant skills. Authorities include, as appropriate, the legally responsible roles and individuals. These security-relevant authorities include security design authorization, security test and operation authorization, and the award of certification, accreditation, or authorization.

PL-2.6 Define the security aspects of infrastructure and services required.

Note: This includes defining the capacity needed for security infrastructure and services, its availability, and its allocation to project tasks. Security infrastructure includes facilities (e.g., Sensitive Compartmented Information Facilities [SCIFs] and isolated networks), specific strength of mechanism mediated access, cross-domain solutions, tools, communication, and information technology assets.

PL-2.7 Plan the security aspects of acquiring materials and enabling system services supplied from outside of the project.

PL-2.8 Generate and communicate a plan for the security aspects of project and technical management and execution, including security reviews that address security considerations.

Note: Security considerations and the planning to address those considerations are captured in a Systems Engineering Management Plan, Software Engineering Management Plans, and similar plans.

References: [ISO 15288, Section 6.3.1.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.3.1.3.2]; [ISO 21827].

PL-3 ACTIVATE THE PROJECT

PL-3.1 Obtain authorization for the security aspects of the project.

PL-3.2 Submit requests and obtain commitments for the necessary resources to perform the security aspects of the project.

PL-3.3 Implement the security aspects of project plans.

References: [ISO 15288, Section 6.3.1.3 c)].

Related Publications: [ISO 12207, Section 6.3.1.3.3]; [ISO 21827].

3.3.2 Project Assessment and Control

The purpose of the *Project Assessment and Control* process is to assess if the plans are aligned and feasible; determine the status of the project, technical, and process performance; and direct execution to help ensure that the performance is within projected budgets according to plans and schedules to satisfy technical objectives.

3.3.2.1 Security Purpose

- To assess if the security aspects of plans and security plans are aligned and feasible
- To determine the state of the project, technical, and process security performance

- To direct execution to help ensure that the security performance is within projected budgets according to plans and schedules to satisfy security and other technical objectives

3.3.2.2 Security Outcomes

- Security aspects of performance measures or assessment results are available.
- Adequacy of security-relevant roles, responsibilities, accountabilities, authorities, and resources is assessed.
- Security aspects of technical progress reviews are performed.
- Deviations in the security aspects of project performance from plans are analyzed.
- Affected stakeholders are informed of the security aspects of project status.
- Corrective action is directed when project performance or achievement is not meeting security-relevant targets.
- Security aspects of project replanning are initiated as necessary.
- Security aspects of project action to progress (or not) from one scheduled milestone or event to the next is authorized.

3.3.2.3 Security Activities and Tasks

PA-1 PLAN FOR PROJECT ASSESSMENT AND CONTROL

PA-1.1 Define the security aspects of the project assessment and control strategy.

Note 1: This includes the planned security assessment methods and time frames as well as necessary security management and technical reviews.

Note 2: Expectations of regulatory, certification, and authorization entities inform the security aspects of the project assessment and control strategy.

References: [ISO 15288, Section 6.3.2.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

PA-2 ASSESS THE PROJECT

PA-2.1 Assess the alignment of the security aspects of project objectives and plans with the project context.

PA-2.2 Assess the security aspects of the management and technical plans against objectives to determine adequacy and feasibility.

PA-2.3 Assess the security aspects of the project and technical status against appropriate plans to determine actual and projected cost, schedule, and performance variances.

PA-2.4 Assess the adequacy of the security-relevant roles, responsibilities, accountabilities, and authorities.

Note: This includes assessment of the adequacy of personnel competencies to perform project roles and accomplish project tasks.

PA-2.5 Assess the security aspects of resource adequacy and availability.

PA-2.6 Assess progress using measured security achievement and security aspects of milestone completion.

Note: This includes collecting and evaluating security-relevant data for labor, material, service costs, and technical performance, as well as other technical data about security objectives. These are compared against security-relevant measures of achievement. This includes conducting effectiveness assessments to determine the adequacy of the evolving system to security requirements.

PA-2.7 Conduct required management and technical reviews, audits, and inspections relevant to the security aspects of the project.

Note: The reviews, audits, and inspections are formal or informal and are conducted to determine the security-relevant readiness to proceed to the next stage or milestone, to help ensure project and technical security objectives are being meet, or to solicit feedback from stakeholders with security concerns.

PA-2.8 Monitor the security aspects of critical processes and new technologies.

Note: This includes identifying and evaluating technology maturity from a security perspective, as well as the feasibility of technology insertion for satisfying security objectives.

PA-2.9 Make recommendations based on security measurement results and other security-relevant project information.

Note: Measurement results are analyzed to identify security-relevant deviations, variations, or undesirable trends from planned values and to make security-relevant recommendations for corrective, preventive, adaptive, additive, or perfective actions.

PA-2.10 Record and provide security status and security findings from the assessment tasks.

PA-2.11 Monitor the security aspects of process execution within the project.

Note: This includes an analysis of process security measures and a review of security-relevant trends with respect to project objectives.

References: [ISO 15288, Section 6.3.2.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Sections 6.3.2.3.1, 6.3.2.3.3]; [ISO 21827].

PA-3 CONTROL THE PROJECT

PA-3.1 Initiate the actions needed to address identified security issues.

PA-3.2 Initiate the necessary security aspects of project replanning.

Note: Replanning is initiated when the security aspects of project objectives or constraints have changed or when security-relevant planning assumptions are shown to be invalid.

PA-3.3 Initiate necessary change actions when there is a contractual change to cost, time, or quality due to the security impact of an acquirer or supplier request.

Note: The security impact is not necessarily obvious in the case where the request is not security-driven or security-oriented.

PA-3.4 Recommend that the project proceed toward the next milestone or event, if justified, based on the achievement of security-relevant milestones or event criteria.

References: [ISO 15288, Section 6.3.2.3 c)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207] 12207, Sections 6.3.2.3.2, 6.3.2.3.4]; [ISO 21827].

3.3.3 Decision Management

The purpose of the *Decision Management* process is to provide a structured, analytical framework for objectively identifying, characterizing, and evaluating a set of alternatives for a decision at any point in the life cycle and select the most beneficial course of action.

[ISO 15288] *Reprinted with permission from IEEE, Copyright IEEE 2015, All rights reserved.*

3.3.3.1 Security Purpose

- To identify, analyze, characterize, and evaluate the security aspects of alternatives for a decision
- To recommend the most beneficial course of security-informed action

3.3.3.2 Security Outcomes

- Security aspects of decisions requiring alternative analysis are identified.
- Security aspects of alternative courses of action are identified and evaluated.
- A preferred security-informed course of action is selected.
- Security aspects of a resolution, of the decision rationale, and of the assumptions are identified.

3.3.3.3 Security Activities and Tasks

DM-1 PREPARE FOR DECISIONS

DM-1.1 Define the security aspects of the decision management strategy.

Note: A decision management strategy includes the identification of security-relevant roles, responsibilities, accountabilities, and authorities. It includes the identification of security-specific decision categories and a prioritization scheme. Security-related decisions often arise as a result of a security effectiveness assessment, a technical trade-off, a security-related problem needing to be solved, an action needed as a response to security risk that exceeds the acceptable threshold, or a new opportunity.

DM-1.2 Identify the security aspects of the circumstances and need for a decision.

DM-1.3 Identify stakeholders with relevant security expertise to support decision-making efforts.

References: [ISO 15288, Section 6.3.3.3 a)].

Related Publications: [ISO 12207, Section 6.3.3.3.1]; [ISO 21827].

DM-2 ANALYZE THE DECISION INFORMATION

DM-2.1 Select and declare the security aspects of the decision management strategy for each decision.

Note: This includes the security-related level of rigor and the data and system analysis needed.

DM-2.2 Determine the desired security outcomes and the measurable security attributes of selection criteria.

Note: The desired value for all quantifiable security criteria and the threshold value(s) beyond which the attribute will be unsatisfactory are determined.

DM-2.3 Identify the security aspects of the trade space and alternatives.

Note: If a large number of alternatives exist, security aspects are to qualitatively screen in order to reduce alternatives to a manageable number for further detailed system analysis.

DM-2.4 Evaluate each alternative against the security criteria.

References: [ISO 15288, Section 6.3.3.3 b)].

Related Publications: [ISO 12207, Section 6.3.3.3.2]; [ISO 21827].

DM-3 MAKE AND MANAGE DECISIONS

DM-3.1 Determine the preferred alternative for each security-informed and security-based decision.

DM-3.2 Record the security-informed or security-based resolution, decision rationale, and assumptions.

DM-3.3 Record, track, evaluate, and report the security aspects of security-informed and security-based decisions.

Note: Security aspects of problems or opportunities and the alternative courses of action that will resolve their outcome – including those with security impacts – are recorded, categorized, and reported.

References: [ISO 15288, Section 6.3.3.3 c)].

Related Publications: [ISO 12207, Section 6.3.3.3.3]; [ISO 21827].

3.3.4 Risk Management

The purpose of the *Risk Management* process is to identify, analyze, treat, and monitor the risks continually.

3.3.4.1 Security Purpose

- To continually identify, analyze, treat, and monitor the risks associated with the uncertainty of achieving security objectives and the effects of security protection efforts on achieving system objectives

3.3.4.2 Security Outcomes

- Security-related risks are identified.
- Security-related risks are analyzed.
- Security-related risk treatments are selected.
- Appropriate security-related risk treatments are implemented.
- Security-related risks are evaluated on an ongoing basis to assess changes in status and progress in treatment.
- Security-related risks are recorded and maintained in the risk profile.

3.3.4.3 Security Activities and Tasks

RM-1 PLAN RISK MANAGEMENT

RM-1.1 Define the security aspects of the risk management strategy.

Note 1: The nature of security risk includes intentional and unintentional casual events, considerations of the intended behaviors and outcomes, functions (security and other functions), and the potential effects of security risk realization. Casual events may be combinations of events in the operational environment and events in the system environment.

Note 2: The security aspects scope of the risk management process, risk management approach, risk criteria, measures, parameters, rating scale, and treatment alternatives are defined. This includes security aspects of the risk management process at all levels of the supply chain (e.g., suppliers, subcontractors) and how they are incorporated into the project risk management process.

Note 3: The strategy can also include those security-relevant issues (e.g., risks with likelihood of occurrence of 1) and opportunities within scope and approach. Opportunity aspects include opportunity criteria, measures, parameters, rating scale, and treatment alternatives.

RM-1.2 Define and record the security context of the risk management process.

Note 1: This includes the identification of security-relevant stakeholders and descriptions of their perspectives, risk categories, and technical and managerial objectives, assumptions, and constraints.

Note 2: Security opportunities provide potential benefits for the system or project. Security contexts consider the security impact of not pursuing an opportunity and the security risk of not achieving the effects provided by the opportunity.

References: [ISO 15288, Section 6.3.4.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 16085]; [ISO 31000].

Related Publications: [ISO 12207, Section 6.3.4.3.1]; [ISO 21827].

RM-2 MANAGE THE RISK PROFILE

RM-2.1 Define and record the security risk thresholds and conditions.

Note: The security risk thresholds define the levels at which the appropriate treatment strategies are considered.

RM-2.2 Establish and maintain the security aspects of the risk profile.

Note: The risk profile records each security risk and opportunity including a description of the security risk or opportunity, a record of the risk or opportunity parameters, the priority based on risk or opportunity criteria, and the risk or opportunity current state, treatment, and contingency strategy. The risk profile is updated when there are changes in an individual security risk or opportunity state.

RM-2.3 Provide the security aspects of the relevant risk profile to stakeholders.

Note: The frequency of communicating the risk profile and its security aspects is determined by project planning.

References: [ISO 15288, Section 6.3.4.3 b)]; [ISO 31000]; [ISO 16085].

Related Publications: [ISO 12207, Section 6.3.4.3.2]; [ISO 21827].

RM-3 ANALYZE RISK

RM-3.1 Identify security risks in the categories described in the risk management context.

Note: Security risks are commonly identified through various security and other analyses, such as safety, assurance, producibility, and performance analyses; technology, architecture, integration, and readiness assessments; measurement reports; and trade-off studies. Additionally, security risks are often identified through the analysis of measures associated with system security goals (e.g., security-relevant Measures of Effectiveness or Measures of Performance).

RM-3.2 Measure each identified security risk.

Note: A common risk measurement is the likelihood of occurrence and consequences as well as the levels of confidence with those measures.

RM-3.3 Evaluate each security risk against its risk thresholds.

RM-3.4 Define and record recommended treatment strategies and measures for each security-relevant risk that exceeds its risk threshold.

References: [ISO 15288, Section 6.3.4.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 31000]; [ISO 16085].

Related Publications: [ISO 12207, Section 6.3.4.3.3]; [ISO 21827].

RM-4 TREAT RISKS THAT EXCEED THEIR RISK THRESHOLD

RM-4.1 Identify recommended alternatives for security risk treatment.

RM-4.2 Define measures for determining the effectiveness of security risk treatments.

RM-4.3 Implement selected security risk treatments.

Note: The implemented alternative should be the one for which the security-relevant stakeholders determine the actions taken will make a security-relevant risk acceptable.

RM-4.4 Coordinate management action for selected security risk treatments.

References: [ISO 15288, Section 6.3.4.3 d)]; [ISO 31000]; [ISO 16085].

Related Publications: [ISO 12207, Section 6.3.4.3.4]; [ISO 21827].

RM-5 MONITOR RISK

RM-5.1 Continually monitor all security-relevant risks and the security risk management context.

Note: Changes with security-relevant risks and their treatments may prompt reevaluation. The initial treatment plans for a security-relevant risk may include preplanned additional actions when risk increases or insufficiently decreases despite treatment.

RM-5.2 Implement and monitor measures to evaluate the effectiveness of security-relevant risk treatments.

RM-5.3 Continually monitor for the emergence of new security-relevant risks and sources of risk throughout the life cycle.

Note: This includes monitoring known changes in adversities.

References: [ISO 15288, Section 6.3.4.3 e)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 31000]; [ISO 16085].

Related Publications: [ISO 12207, Section 6.3.4.3.5]; [ISO 21827].

3.3.5 Configuration Management

The purpose of the *Configuration Management* process is to manage system and system elements and configurations over the life cycle.

3.3.5.1 Security Purpose

- To incorporate security considerations to securely manage system and system elements and configurations over the life cycle

3.3.5.2 Security Outcomes

- System element configurations are securely managed.
- Security aspects of configuration baselines are established.
- Changes to items under configuration management are securely controlled.
- Security aspects of configuration status information are available.
- Security aspects of required configuration audits are completed.
- Security aspects of system releases are approved.

3.3.5.3 Security Activities and Tasks

CM-1 PREPARE FOR CONFIGURATION MANAGEMENT

CM-1.1 Define a secure configuration management strategy.

Note: These include:

- Security-relevant roles, responsibilities, accountabilities, and authorities
- Criteria for the secure management of changes to items under configuration management, including dispositions, access, release, and control
- Security considerations, criteria, and constraints for the locations, conditions, and environment of storage
- Criteria or events for commencing secure configuration control and securely maintaining baselines of evolving configurations
- Security aspects of the audit strategy and the responsibilities for assessing continual integrity and security of the configuration definition information
- Criteria and constraints for secure change management, planned configuration control boards and security configuration control boards, regulatory and emergency change requests, and procedures for secure change management
- Secure coordination among stakeholders, acquirers, suppliers, supply chain, and other interacting organizations

CM-1.2 Define the secure archive and retrieval approach for configuration items, configuration management artifacts, and data.

Note: This includes rules governing secure retention, access, and use.

References: [ISO 15288, Section 6.3.5.3 a)]; [ISO 10007]; [IEEE 828]; [EIA 649C].

Related Publications: [ISO 12207, Sections 6.3.5.3.1, 7.2.2.3.1]; [ISO 21827].

CM-2 PERFORM CONFIGURATION IDENTIFICATION

CM-2.1 Identify the security aspects of system elements and artifacts that need to be under configuration management.

CM-2.2 Identify the security aspects of the configuration data to be managed.

CM-2.3 Establish the security aspects of identifiers for items under configuration management.

CM-2.4 Define the security aspects of baselines through the life cycle.

CM-2.5 Obtain applicable stakeholder agreement of the security aspects to establish a baseline.

CM-2.6 Approve and track security aspects of system or system element releases.

Note 1: The security aspects of a release are security-relevant considerations of authorization of the use of a system or system element for a specific purpose with or without security-relevant restrictions. Examples are releases for tests or operational use.

Note 2: Releases generally include a set of changes made through the Technical Processes. Release approval generally includes acceptance of the verified and validated changes and any impacts to security of the changes.

References: [ISO 15288, Section 6.3.5.3 b)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Sections 6.3.5.3.2, 7.2.2.3.2]; [ISO 21827].

CM-3 PERFORM CONFIGURATION CHANGE MANAGEMENT

CM-3.1 Identify and record the security aspects of requests for change and requests for variance.

Note 1: This includes requests for deviation, waiver, or concession.

Note 2: Change or variance can be based on reasons other than security or without an obvious relevance to security.

CM-3.2 Determine the security aspects of action to coordinate, evaluate, and disposition requests for change or requests for variance.

Note: The security aspects identified are coordinated and evaluated across all impacted performance and effectiveness evaluation criteria, as well as the criteria of project plans, cost, benefits, risks, quality, and schedule.

CM-3.3 Submit requests for security review and approval.

Note: Control boards may or may not be security focused. For a non-security control board activity, security should be reviewed to verify that there are no security aspects to a request.

CM-3.4 Track and manage the security aspects of approved changes to the baseline, requests for change, and requests for variance.

References: [ISO 15288, Section 6.3.5.3 c)].

Related Publications: [ISO 12207, Sections 6.3.5.3.2, 7.2.2.3.3]; [ISO 21827].

CM-4 PERFORM CONFIGURATION STATUS ACCOUNTING

CM-4.1 Develop and maintain security-relevant configuration management status information for system elements, baselines, approved changes, and releases.

Note: The information includes security certification, accreditation, authorization, or approval decisions for a system, system element, baseline, or release.

CM-4.2 Capture, store, and report security-relevant configuration management data.

References: [ISO 15288, Section 6.3.5.3 d)].

Related Publications: [ISO 12207, Section 7.2.2.3.4]; [ISO 21827].

CM-5 PERFORM CONFIGURATION EVALUATION

CM-5.1 Identify the need for secure configuration and configuration management verification activities and audits.

CM-5.2 Verify that the product or service configuration meets the security-relevant configuration requirements.

Note: This is performed by comparing security requirements, constraints, and waivers (variances) with the results of formal verification activities.

CM-5.3 Monitor the secure incorporation of approved configuration changes.

CM-5.4 Perform configuration and configuration management security verification activities and audits to establish the security aspects of product baselines.

Note: This includes the security aspects of the functional configuration audit (FCA) that are focused on functional and performance capabilities and the security aspects of the physical configuration audit (PCA) that are focused on system conformance to operational and configuration information items.

CM-5.5 Record the security aspects of the configuration management audit and other configuration evaluation results and disposition action items.

References: [ISO 15288, Section 6.3.5.3 e)].

Related Publications: [ISO 12207, Section 7.2.2.3.5]; [ISO 21827].

3.3.6 Information Management

The purpose of the *Information Management* process is to generate, obtain, confirm, transform, retain, retrieve, disseminate, and dispose of information to designated stakeholders.

3.3.6.1 Security Purpose

- To address the security aspects of information management

3.3.6.2 Security Outcomes

- Security-relevant information to be managed is identified.
- Security protections for information are identified.
- Security aspects of information representations are defined.
- Information is securely managed.
- Security aspects of information status are identified.
- Information is available to designated stakeholders in a secure manner.

3.3.6.3 Security Activities and Tasks

IM-1 PREPARE FOR INFORMATION MANAGEMENT

IM-1.1 Define the security aspects of the strategy for information management.

Note: The security aspects include stakeholder, technical, and other information. These aspects address security, privacy, and intellectual property concerns.

IM-1.2 Define the security aspects of the items of information that will be managed.

IM-1.3 Designate authorities and responsibilities for the security aspects of information management.

Note: Due regard is paid to legislation, security, and privacy (e.g., ownership, agreement restrictions, rights of access, data rights, and intellectual property). Where restrictions or constraints apply, information is identified accordingly. Staff with knowledge of such items of information are informed of their security-relevant obligations and responsibilities.

IM-1.4 Define the security aspects of the content, formats, structure, and strengths of protection for information items.

Note 1: The security aspects apply to information while at rest (i.e., persistent or non-persistent storage) and while in transit between a source/point of origin and destination.

Note 2: The security aspects are informed by criteria in applicable laws, policies, directives, regulations, and patents.

IM-1.5 Define the security aspects of information maintenance actions.

References: [ISO 15288, Section 6.3.6.3 a)].

Related Publications: [ISO 12207, Section 6.3.6.3.1]; [ISO 21827].

IM-2 PERFORM INFORMATION MANAGEMENT

IM-2.1 Securely obtain, develop, or transform the identified information items.

IM-2.2 Securely maintain information items and their storage records, and record the security status of information.

IM-2.3 Securely publish, distribute, or provide access to information and information items to designated stakeholders.

IM-2.4 Securely archive designated information.

Note: The media, location, and protection of the information are selected in accordance with the specified storage and retrieval periods, agreements, legislation, and organizational security policy.

IM-2.5 Securely dispose of unwanted, invalid, or unvalidated information.

References: [ISO 15288, Section 6.3.6.3 b)].

Related Publications: [ISO 12207, Section 6.3.6.3.2]; [ISO 21827].

3.3.7 Measurement

The purpose of the *Measurement* process is to collect, analyze, and report objective data and information to support effective management and demonstrate the quality of the products, services, and processes.

3.3.7.1 Security Purpose

- To collect, analyze, and report security-relevant data and information to support effective management and demonstrate the quality of the products, services, and processes

3.3.7.2 Security Outcomes

- Security-relevant information needs are identified.

- An appropriate set of security measures are identified or developed based on security-relevant information needs and information security protection needs.
- Required data is securely managed.
- Security-relevant data is analyzed and the results interpreted.
- Measurement results provide objective information that supports security-relevant decisions.

3.3.7.3 Security Activities and Tasks

MS-1 PREPARE FOR MEASUREMENT

MS-1.1 Define the security aspects of the measurement strategy.

MS-1.2 Describe the characteristics of the organization that are relevant to security measurement.

MS-1.3 Identify and prioritize security-relevant information needs.

Note: The needs are based on protection objectives, identified security risks, and other security-relevant items related to project decisions.

MS-1.4 Select and specify measures that satisfy security-relevant information needs.

MS-1.5 Define procedures for the collection, analysis, access, and reporting of security-relevant data.

MS-1.6 Define security-relevant criteria for evaluating the information items and the measurement process.

Note: All criteria for a security-relevant information item are security-relevant.

MS-1.7 Identify the security aspects for enabling the systems or services needed to support measurement.

MS-1.8 Identify and plan for enabling the systems or services needed to support the security aspects of measurement.

MS-1.9 Obtain or acquire access to the security aspects of enabling systems or services to be used in measurement.

References: [ISO 15288, Section 6.3.7.3 a)]; [ISO 9001]; [ISO 15939].

Related Publications: [ISO 12207, Section 6.3.7.3.1].

MS-2 PERFORM MEASUREMENT

MS-2.1 Integrate procedures for the generation, collection, analysis, and reporting of security-relevant data into the relevant processes.

MS-2.2 Integrate procedures for the secure generation, collection, analysis, and reporting of data into the relevant processes.

MS-2.3 Collect, store, and verify security-relevant data.

MS-2.4 Securely collect, store, and verify data.

MS-2.5 Analyze security-relevant data, and develop security-relevant information items.

MS-2.6 Record security measurement results and inform the measurement users.

Note: Security measurement results are provided to stakeholders and project personnel to support decision-making, risk management, and to initiate corrective actions and improvements.

References: [ISO 15288, Section 6.3.7.3 b)]; [ISO 9001]; [ISO 15939].

Related Publications: [ISO 12207, Sections 6.3.7.3.2, 6.3.7.3.3].

3.3.8 Quality Assurance

The purpose of the *Quality Assurance* process is to help ensure the effective application of the organization's *Quality Management* process to the project.

3.3.8.1 Security Purpose

- To help ensure the effective application of the organization's *Quality Management* process to the security aspects of the project

3.3.8.2 Security Outcomes

- Security aspects of quality assurance procedures, including security criteria and methods for quality assurance evaluations, are implemented.
- Evaluations of the products, services, and processes of the project are performed in a manner consistent with security quality management policies, procedures, and requirements.
- Security results of evaluations are provided to relevant stakeholders.
- Security-relevant incidents are resolved.
- Prioritized security-relevant problems are treated.

3.3.8.3 Security Activities and Tasks

QA-1 PREPARE FOR QUALITY ASSURANCE

QA-1.1 Define the security aspects of the quality assurance strategy.

Note: The security aspects are informed by and consistent with the quality management policies, objectives, and procedures and include:

- Project security quality assurance procedures
- Security roles, responsibilities, accountabilities, and authorities
- Security activities appropriate to each life cycle process
- Security activities appropriate to each supplier (including subcontractors)
- Required security-oriented verification, validation, monitoring, measurement, inspection, and test activities specific to the product or service
- Security criteria for product or service acceptance

QA-1.2 Establish the independence of security quality assurance from other life cycle processes.

References: [ISO 15288, Section 6.3.8.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3].

Related Publications: [ISO 12207, Section 7.2.3.3.1].

QA-2 PERFORM PRODUCT OR SERVICE EVALUATIONS

QA-2.1 Evaluate products and services for conformance to established security criteria, contracts, standards, and regulations.

QA-2.2 Perform the security aspects of verification and validation on the outputs of the life cycle processes to determine conformance to specified requirements.

References: [ISO 15288, Section 6.3.8.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 7.2.3.3.2].

QA-3 PERFORM PROCESS EVALUATIONS

QA-3.1 Evaluate project life cycle processes for conformance to established security quality criteria.

QA-3.2 Evaluate tools and environments that support or automate the process for conformance to established security quality criteria.

QA-3.3 Evaluate supplier processes for conformance to process security requirements.

Note: Consider items such as the security aspects of development environments, process measures that suppliers are required to provide, or a risk process that suppliers are required to use.

References: [ISO 15288, Section 6.3.8.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 7.2.3.3.3].

QA-4 MANAGE QUALITY ASSURANCE RECORDS AND REPORTS

QA-4.1 Create records and reports related to the security aspects of quality assurance activities.

QA-4.2 Securely maintain, store, and distribute records and reports.

QA-4.3 Identify the security aspects of incidents and problems associated with product, service, and process evaluations.

References: [ISO 15288, Section 6.3.8.3 d)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 7.2.3.3.4].

QA-5 TREAT INCIDENTS AND PROBLEMS

QA-5.1 Record, analyze, and classify the security aspects of incidents.

Note: Incidents are grouped (classified) by criteria such as type, scope, and effect.

QA-5.2 Resolve the security aspects of incidents, or elevate the security aspects of incidents to problems.

QA-5.3 Record, analyze, and classify the security aspects of problems.

QA-5.4 Track the security aspects of the prioritization and implementation of problem treatment.

Note: This includes both security-driven problem treatment and the security aspects of general problem treatment.

QA-5.5 Note and analyze the security aspects of incidents and problems.

QA-5.6 Inform stakeholders of the status of the security aspects of incidents and problems.

QA-5.7 Track the security aspects of incidents and problems to closure.

References: [ISO 15288, Section 6.3.8.3 e)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 24748-1].

Related Publications: None.

3.4 TECHNICAL PROCESSES

This section contains the *Technical Processes* from [ISO 15288] with security-related considerations and contributions.

3.4.1 Business or Mission Analysis

The purpose of the *Business or Mission Analysis* process is to define the overall strategic problem or opportunity, characterize the solution space, and determine potential solution class(es) that can address a problem or take advantage of an opportunity.

3.4.1.1 Security Purpose

- To define the security aspects related to the strategic problems or opportunities
- To identify the security objectives, concerns, and constraints that inform the potential solution classes

3.4.1.2 Security Outcomes

- Security aspects of the strategic problem or opportunity space are defined.
- Security aspects of the solution space are characterized.
- The definition of the preliminary operational concepts and other concepts in the life cycle stages are informed by the security aspects of the problem or opportunity space.
- Alternative solution classes are analyzed considering identified security aspects.
- Selection of the preferred alternative solution class(es) is informed by the security aspects of the solution space.
- Enabling systems or services needed for the security aspects of business or mission analysis are available.
- Traceability of the security aspects of the strategic problems and opportunities to the preferred alternative solution classes is established.

3.4.1.3 Security Activities and Tasks

BA-1 PREPARE FOR BUSINESS OR MISSION ANALYSIS

BA-1.1 Identify the security aspects for enabling systems or services needed to support business or mission analysis.

BA-1.2 Identify and plan for enabling systems or services needed to support the security aspects of business or mission analysis.

BA-1.3 Obtain or acquire access to the security aspects of enabling systems or services to be used in business or mission analysis.

References: [ISO 15288, Section 6.4.1.3 a)].

Related Publications: None.

BA-2 DEFINE THE PROBLEM OR OPPORTUNITY SPACE

BA-2.1 Analyze the problems or opportunities in the context of the security-relevant trade space factors.

Note: The security-relevant trade space factors are analyzed within the context of all factors, including factors related to loss tolerances. The results of the analyses inform decisions on the suitability and feasibility of alternative options to be pursued.

BA-2.2 Define the security aspects of the mission, business, or operational problem or opportunity to be addressed by the solution class(es).

Note: Information is elicited from stakeholders to acquire an understanding of the mission, business, or operational problem or opportunity from a system security perspective. Security aspects include security objectives, concerns, and constraints.

References: [ISO 15288, Section 6.4.1.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: None.

BA-3 CHARACTERIZE THE SOLUTION SPACE

BA-3.1 Define the security aspects of the preliminary operational concepts and other concepts in life cycle stages.

Note 1: Security operational concepts include modes of secure operation, security-related operational scenarios and use cases, and secure usage within a mission area or line of business.

Note 2: Security aspects are integrated into the life cycle concepts and used to support feasibility analysis and the evaluation of candidate alternative solution classes.

BA-3.2 Identify the security aspects of the alternative solution classes.

References: [ISO 15288, Section 6.4.1.3 c)]; [ISO 42010]; [ISO 24748-1].

Related Publications: None.

BA-4 EVALUATE ALTERNATIVE SOLUTION CLASSES

BA-4.1 Assess each alternative solution class while considering the identified security aspects.

BA-4.2 Select the preferred alternative solution class (or classes) based on the identified security aspects, trade space factors, and other criteria defined by the organization.

BA-4.3 Provide security-relevant feedback to strategic level life cycle concepts to reflect the selected solution class(es).

References: [ISO 15288, Section 6.4.1.3 d)]; [ISO 42010]; [ISO 24748-1].

Related Publications: None.

BA-5 MANAGE THE BUSINESS OR MISSION ANALYSIS

BA-5.1 Maintain traceability of the security aspects of business or mission analysis.

Note: Bidirectional traceability is maintained between identified security aspects and supporting security data associated with the problems and opportunities, proposed solution class or classes, and organizational strategy.

BA-5.2 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.1.3 e)]; [ISO 42010]; [ISO 24748-1].

Related Publications: None.

3.4.2 Stakeholder Needs and Requirements Definition

The purpose of the *Stakeholder Needs and Requirements Definition* process is to define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.

3.4.2.1 Security Purpose

- To identify the protection needs associated with the stakeholder needs and requirements for a system that can protect the capabilities needed by users and other stakeholders in a defined environment

3.4.2.2 Security Outcomes

- Security-relevant stakeholders of the system are identified.
- Security concerns of stakeholders are identified.
- Required characteristics and context for the secure use of capabilities for system life cycle concepts in system life cycle stages are defined.
- Stakeholder assets and asset classes are identified.
- Adversity presented by the environment is characterized.
- Asset protection priorities are determined.
- Stakeholder protection needs are defined.
- Security-driven and security-informed constraints on a system are identified.
- Prioritized stakeholder protection needs are transformed into stakeholder requirements.
- Security-oriented performance measures and quality characteristics are defined.
- Stakeholder agreement that their protection needs and expectations are adequately reflected in the requirements is achieved.
- Enabling systems or services needed for the security aspects of stakeholder needs and requirements definition are available.
- Traceability of stakeholder requirements to stakeholders and their protection needs is established.

3.4.2.3 Security Activities and Tasks

SN-1 PREPARE FOR STAKEHOLDER NEEDS AND REQUIREMENTS DEFINITION

SN-1.1 Identify the stakeholders and their security concerns.

Note 1: All stakeholders have security concerns, whether implicit or explicit.

Note 2: This includes stakeholders who represent milestone decision authority, regulatory, certification, authorization, acceptance, and similar organizations with specific security-related decision-making authority and responsibilities.

SN-1.2 Define the stakeholder protection needs and requirements definition strategy.

Note: The strategy includes addressing how consensus about protection needs and requirements is to be achieved among stakeholders with opposing interests.

SN-1.3 Identify the security aspects for enabling systems or services needed to support stakeholder needs and requirements definition.

SN-1.4 Identify and plan for enabling systems or services needed to support the security aspects of stakeholder needs and requirements definition.

SN-1.5 Obtain or acquire access to the security aspects of enabling systems or services to be used in stakeholder needs and requirements definition.

References: [ISO 15288, Section 6.4.2.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.4.1.3.1]; [ISO 21827].

SN-2 DEVELOP THE OPERATIONAL AND OTHER LIFE CYCLE CONCEPTS

SN-2.1 Define a representative set of scenarios to identify required protection capabilities and security measures that correspond to anticipated operational and other life cycle concepts.

Note: The scenarios reflect how the system is intended to behave in the intended operational environments. Scenarios also help to identify security-driven changes to life cycle concepts.

SN-2.2 Characterize the security aspects of the operational environments and the intended users.

Note 1: This includes distinguishing what is and is not known about adversity within the operational environments.

Note 2: This includes the trust expectations for users to address insider threat concerns. If a user security aspect cannot be obtained or there is uncertainty about the trust of users, it will significantly drive design and the operational procedure to complement the design.

SN-2.3 Identify the interactions among entities (e.g., personnel, enabling and other interfacing systems) and the system and security-related factors affecting the interactions.

Note: The interactions among entities and the system and the factors affecting the interactions need to be understood to inform engineering efforts. Factors influencing the interactions include the environment of the system of interest and any system of systems the system of interest belongs to, as well as the characterization of the entities with which the system interacts.

SN-2.4 Identify the security-related constraints on a system solution.

References: [ISO 15288, Section 6.4.2.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 18152]; [ISO 25060]; [ISO 25063]; [ISO 29148].

Related Publications: [ISO 9241]; [ISO 21827]; [ISO 25010].

SN-3 DEFINE STAKEHOLDER NEEDS

SN-3.1 Define the rules capturing authorized and intended interactions, behaviors, and outcomes.

Note: The life cycle concepts and their context inform the rules.

SN-3.2 Identify stakeholder assets and asset classes.

SN-3.3 Identify loss concerns for each identified asset and each asset class.

SN-3.4 Prioritize assets based on the adverse consequence of asset loss.

SN-3.5 Determine adversities present in the environment.

Note: Environments that expose the system to potential adversities can include test, operational, maintenance, and logistical environments. The adversities need to be avoided when possible and protected against otherwise.

SN-3.6 Identify stakeholder protection needs.

SN-3.7 Prioritize and down-select the stakeholder protection needs.

SN-3.8 Record the stakeholder protection needs and rationale.

References: [ISO 15288, Section 6.4.2.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 25063].

Related Publications: [ISO 21827]; [ISO 18152]; [ISO 25010].

SN-4 TRANSFORM STAKEHOLDER NEEDS INTO STAKEHOLDER REQUIREMENTS

SN-4.1 Identify the security-related constraints on a system solution.

SN-4.2 Define stakeholder requirements in a manner consistent with security aspects and protection needs.

References: [ISO 15288, Section 6.4.2.3 d)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 25030].

Related Publications: [ISO 12207, Section 6.4.1.3.2]; [ISO 21827]; [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 27034-1].

SN-5 ANALYZE STAKEHOLDER NEEDS AND REQUIREMENTS

SN-5.1 Analyze the set of stakeholder requirements with respect to the protection needs.

Note: The stakeholder requirements are analyzed to determine if the protection needs are accurately and comprehensively expressed in both individual requirements and the set of requirements. Potential analysis characteristics include that the requirements: (1) are necessary, complete, succinct, and implementation-free, and (2) comprehensively address the protection needs.

SN-5.2 Define security-relevant performance and assurance measures that enable the assessment of technical achievement and their relative criticality.

Note: Determining the relative criticality of measures captures technical achievements and reflects stakeholder priorities.

SN-5.3 Provide feedback to applicable stakeholders from the analyzed requirements to validate that their protection needs and expectations have been adequately captured and expressed.

SN-5.4 Resolve stakeholder requirements issues related to protection needs.

Note: Any change to stakeholder requirements signifies a need to reassess protection needs and determine if any subsequent changes are required.

References: [ISO 15288, Section 6.4.2.3 e)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 15939]; [ISO 29148]; [INCOSE10].

Related Publications: [ISO 12207, Section 6.4.1.3.3]; [ISO 21827].

SN-6 MANAGE THE STAKEHOLDER NEEDS AND REQUIREMENTS DEFINITION

SN-6.1 Obtain explicit agreement that the stakeholder requirements satisfactorily address protection needs.

SN-6.2 Record asset protection data.

SN-6.3 Maintain traceability between stakeholder protection needs and stakeholder requirements.

SN-6.4 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.2.3 f)].

Related Publications: [ISO 12207, Sections 6.4.1.3.4, 6.4.1.3.5]; [ISO 21827].

3.4.3 System Requirements Definition

The purpose of the *System Requirements Definition* process is to transform the stakeholder, user-oriented view of desired capabilities into a technical view of a solution that meets the operational needs of the user.

3.4.3.1 Security Purpose

- To provide an accurate and complete representation of stakeholder protection needs (as expressed in the stakeholder requirements) in the system requirements

3.4.3.2 Security Outcomes

- Security aspects of the system description – including system interfaces, functions, and boundaries for a system solution – are defined.
- Security-relevant system requirements and security-driven design constraints are defined.
- Security performance measures are defined.
- Security aspects of the system requirements are analyzed.
- Enabling systems or services needed for the security aspects of the system requirements definition are available.
- Traceability of the security aspects of system requirements and associated security-relevant constraints to stakeholder requirements is established.

3.4.3.3 Security Activities and Tasks

SR-1 PREPARE FOR SYSTEM REQUIREMENTS DEFINITION

SR-1.1 Define the security aspects of the intended behavior and outcomes at the functional boundary of the system.

Note: The intended behavior and security properties to be realized at the functional boundary consider the characteristics of the capability provided or used, the characteristics of the entities that interact with the system of interest at the functional boundary, and the associated assurance needs.

SR-1.2 Define the security domains of the system and their correlation to the functional boundaries of the system.

SR-1.3 Define the security aspects of the system requirements definition strategy.

SR-1.4 Identify the security aspects for enabling systems or services needed to support system requirements definition.

SR-1.5 Identify and plan for enabling systems or services needed to support the security aspects of system requirements definition.

SR-1.6 Obtain or acquire access to the security aspects of enabling systems or services to be used in system requirements definition.

References: [ISO 15288, Section 6.4.3.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

SR-2 DEFINE SYSTEM REQUIREMENTS

SR-2.1 Define each security function that the system is required to perform.

Note: Security functions are defined for all system states, modes, and conditions of system operation and use, including the associated transitions between system states and modes. Security functions include those oriented to delivery of capability and the ability of the system to execute while preserving its inherent security characteristics.

SR-2.2 Define the security aspects of each function that the system is required to perform.

Note: This includes the need for other system functions to be non-interfering (see D.4.1).

SR-2.3 Define necessary security-driven implementation constraints.

Note: Security-driven constraints on the system are from adversity, uncertainty, and risk, considering performance objectives and assurance needs. These constraints are informed by stakeholder requirements, the system architecture definition, and solution limitations across the life cycle.

SR-2.4 Define necessary constraints on security implementation.

Note: Constraints on security implementation are to satisfy expectations for non-security capability and performance.

SR-2.5 Define system security requirements and rationale.

Note: System security requirements include security capability and functional requirements, security performance and effectiveness requirements, security assurance requirements, and implementation constraints (SR-2.3 and SR-2.4 outcomes expressed as requirements).

SR-2.6 Apply security metadata to the system security requirements.

Note: Metadata enables identification and traceability to support analysis of completeness and consistency to determine security impact when requirements change.

References: [ISO 15288, Section 6.4.3.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3]; [ISO 29148]; [ISO 25030].

Related Publications: [ISO 12207, Section 6.4.2.3.1]; [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 21827]; [ISO 27034-1].

SR-3 ANALYZE SYSTEM REQUIREMENTS

SR-3.1 Analyze the complete set of system requirements in consideration of security concerns.

Note: Requirements are analyzed to ensure that individual and combinations of requirements fully and properly capture security protection and security-constraint considerations. Rationale is captured to support analysis conclusions and provides a basis to conclude that the analysis has the proper perspective and is fully aware of assumptions made. See Appendix C.

SR-3.2 Define security-driven performance and assurance measures that enable the assessment of technical achievement.

SR-3.3 Provide feedback from the analyzed system requirements to applicable stakeholders for security-relevant reviews.

SR-3.4 Resolve system requirements security issues.

References: [ISO 15288, Section 6.4.3.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 15939]; [ISO 29148]; [INCOSE10].

Related Publications: [ISO 12207, Section 6.4.2.3.2]; [ISO 21827].

SR-4 MANAGE THE SYSTEM REQUIREMENTS

SR-4.1 Obtain explicit agreement that system requirements express protection needs.

SR-4.2 Record key security-related system requirement decisions and the rationale.

SR-4.3 Maintain traceability of system requirements to their security-relevant aspects.

Note: The traceability of system requirements to protection needs; stakeholder requirements; architecture elements; interface definitions; analysis results; verification methods; and all allocated, decomposed, and *derived requirements* (in their system, system element, security protection, and security-driven constraint forms); risk and loss tolerance; and assurance and trustworthiness objectives is maintained.

SR-4.4 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.3.3 d)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

3.4.4 System Architecture Definition

The purpose of the *System Architecture Definition* process is to generate system architecture alternatives, to select one or more alternative(s) that frame stakeholder concerns and meet system requirements, and to express this in a set of consistent views and models.

3.4.4.1 Security Purpose

- To generate the architectural concepts and properties of system architecture alternatives for the system protection capability that frame stakeholder protection concerns and meet system requirements
- To express them in a set of consistent views and models

- To provide the security aspects used to select one or more architecture alternatives

3.4.4.2 Security Outcomes

- The problem space is refined with respect to key stakeholder security concerns.
- Alignment of the architecture with applicable security policies, directives, objectives, and constraints is achieved.
- Concepts, properties, characteristics, behaviors, functions, and constraints that are significant to security-relevant architecture decisions about the system are allocated to architectural entities.
- Identified stakeholder protection concerns are addressed by the system architecture.
- Traceability of the security aspects of system architecture elements to key architecturally relevant stakeholder and system requirements is established.
- Security aspects of architecture views and models of the system are developed.
- Security aspects of system elements, their interactions, and their interfaces are defined.

3.4.4.3 Security Activities and Tasks

AR-1 PREPARE FOR SYSTEM ARCHITECTURE DEFINITION

AR-1.1 Define the security aspects of the system architecture definition strategy.

AR-1.2 Identify the set of existing security-relevant architectures or reference architectures that may have direct applicability and are to be used as guiding oversight.

AR-1.3 Establish the security aspects of the architecture description framework(s), viewpoints, and modeling templates to be used throughout the system architecture definition effort.

AR-1.4 Establish security-specific viewpoints and modeling templates to be used throughout the system architecture definition effort.

AR-1.5 Determine the security evaluation objectives and criteria with respect to the concerns of key stakeholders.

AR-1.6 Determine security evaluation methods and integrate with evaluation objectives and criteria.

AR-1.7 Collect and review security evaluation-related information.

AR-1.8 Identify the security aspects for enabling systems or services needed to support system architecture definition.

AR-1.9 Identify and plan for enabling systems or services needed to support the security aspects of system architecture definition.

AR-1.10 Obtain or acquire access to the security aspects of enabling systems or services to be used in system architecture definition.

References: [ISO 15288, Section 6.4.4.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 42010]; [ISO 42020].

Related Publications: [ISO 21827].

AR-2 CREATE THE SYSTEM ARCHITECTURE CANDIDATE(S)

AR-2.1 Establish the security aspects of architecture objectives and critical success criteria.

AR-2.2 Synthesize potential trustworthy secure solution(s) in the solution space.

AR-2.3 Characterize aspects of trustworthy secure solutions and the trade space.

AR-2.4 Formulate trustworthy secure candidate architecture(s).

AR-2.5 Capture trustworthy secure architecture concepts and properties.

AR-2.6 Relate the candidate architecture(s) to other architectures and relevant affected entities to help ensure the consistency of trustworthy secure architecture concepts and properties.

AR-2.7 Coordinate the secure use of the candidate architecture(s) by intended users.

AR-2.8 Develop the security aspects of the models and views of the candidate architecture(s).

Note: The following are typical considerations to define the security aspects of the system context and boundaries in terms of interfaces and interactions between entities:

- Definition of the system security context and security boundaries in terms of interfaces and interactions with external entities
- The identification of architectural entities and relationships between entities that address key stakeholder protection concerns and system security requirements
- The allocation of security concepts, security properties, security characteristics, secure behaviors, security functions, or security constraints that are significant to architecture decisions of the system to architectural entities
- Composition of views from the models in accordance with identified viewpoints to express how the architecture addresses stakeholder protection concerns and meets stakeholder and system security requirements
- Harmonization of the architecture models and views

AR-2.9 Coordinate secure use of the architecture by intended users.

References: [ISO 15288, Section 6.4.4.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 42010]; [ISO 42020].

Related Publications: [ISO 21827].

AR-3 EVALUATE THE SYSTEM ARCHITECTURE CANDIDATE(S)

AR-3.1 Analyze trustworthy secure architecture concepts and properties, and assess the value of the architecture in meeting stakeholder security protection concerns.

AR-3.2 Characterize the candidate architecture(s) based on trustworthy secure analysis results.

AR-3.3 Formulate security-relevant evaluation findings and recommendations.

AR-3.4 Capture and communicate security-relevant evaluation results.

AR-3.5 Relate the architecture to the other architectures and to relevant affected entities to help ensure consistency in the trustworthy secure system architecture.

References: [ISO 15288, Section 6.4.4.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 42010]; [ISO 42020].

Related Publications: [ISO 21827].

AR-4 MANAGE THE RESULTS OF SYSTEM ARCHITECTURE DEFINITION

AR-4.1 Obtain agreement on the security aspects of the architecture.

AR-4.2 Record key security-relevant system architecture decisions and the rationale.

AR-4.3 Maintain the traceability of the security aspects of the system architecture.

AR-4.4 Provide the security-relevant artifacts that have been selected for baselines.

AR-4.5 Provide support to organizational architecture governance and architecture management efforts.

References: [ISO 15288, Section 6.4.4.3 f)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 42010]; [ISO 42020].

Related Publications: [ISO 21827].

3.4.5 Design Definition

The purpose of the *Design Definition* process is to provide sufficient detailed data and information about the system and its elements to realize the solution in accordance with the system requirements and architecture.

3.4.5.1 Security Purpose

- To provide sufficient detailed data and information about the security aspects of the system and its elements to realize a trustworthy secure solution in accordance with the system requirements and architecture

3.4.5.2 Security Outcomes

- Security aspects of design alternatives for system elements are assessed.
- System requirements are allocated to address their security aspects.
- Security interfaces and security aspects of interfaces between system elements composing the system are defined.
- Security design characteristics of each system element are defined.
- Enabling systems or services for the security aspects of design definition are available.
- Traceability of security design characteristics is established.

3.4.5.3 Security Activities and Tasks

DE-1 PREPARE FOR DESIGN DEFINITION

DE-1.1 Establish the trustworthy secure aspects of the design definition strategy.

DE-1.2 Determine the security technologies required for each system element composing the system.

DE-1.3 Identify the security concerns associated with each technology required for each system element.

Note 1: This includes the security concerns due to vulnerability within or enabled by the supply chains involved with acquisition of the technologies.

Note 2: The concerns may have associated risks to record and track.

DE-1.4 Determine the necessary security and trustworthiness categories of system characteristics represented in the design.

Note: Such characteristics include applying foundational security design principles and concepts with the necessary rigor to achieve target levels of assurance.

DE-1.5 Define the principles for trustworthy secure evolution of the system design.

DE-1.6 Identify the security aspects for enabling systems or services needed to support design definition.

DE-1.7 Identify and plan for enabling systems or services needed to support the security aspects of design definition.

DE-1.8 Obtain or acquire access to the security aspects of enabling systems or services to be used in design definition.

References: [ISO 15288, Section 6.4.5.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

DE-2 CREATE THE SYSTEM DESIGN

DE-2.1 Allocate security requirements to system elements.

Note: This allocates the security aspects of architecture, behavior, and constraints to the system design.

DE-2.2 Transform security-relevant architectural entities and relationships into design elements.

DE-2.3 Transform secure architectural characteristics into trustworthy secure design characteristics.

Note 1: The transformation applies the architectural, trust, and security design principles in successively finer-grained contexts to express the security design characteristics for the constituent components of architectural entities. Security design characteristics apply to security functional capabilities.

Note 2: The characteristics include or reflect the expected level of assurance.

DE-2.4 Define the necessary trustworthy secure design enablers.

Note: Trustworthy secure design enablers include standards, specifications, patterns, models for security policy, security protocols, strength of mechanism, cryptographic algorithms, adversarial threat actors, and functional behaviors and interactions.

DE-2.5 Examine trustworthy secure design alternatives.

Note: Assess the feasibility of each design alternative to minimize susceptibility, exposure, vulnerability, and hazard based on the allocation of system characteristics.

DE-2.6 Refine or define the security aspects of interfaces between system elements and with external entities.

Note: The details of the defined interfaces are refined to capture additional details provided by the security aspects of the design. In addition, the interfaces, interconnections, behavior, and interactions for components within the system of interest are identified, as are the security and security-driven design constraints applied on all interfaces, interactions, and behavior between components of the system of interest.

DE-2.7 Develop the security aspects of design artifacts.

Note: Design artifacts include general and security-specific specifications, data sheets, databases, and documents.

DE-2.8 Capture the security aspects of the design.

References: [ISO 15288, Section 6.4.5.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Sections 6.4.3.3.1, 7.1.4.3.1]; [ISO 27034-1]; [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 21827].

DE-3 EVALUTE THE SYSTEM DESIGN

DE-3.1 Analyze each system design alternative against criteria developed from expected trustworthy secure design properties and characteristics.

DE-3.2 Assess each system design alternative for how well it meets stakeholder protection needs and the security aspects of the system requirements.

DE-3.3 Combine the security analyses and assessments in the overall evaluation to select a preferred design solution.

References: [ISO 15288, Section 6.4.5.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.4.3.3.2]; [ISO 27034-1]; [ISO 21827].

DE-4 MANAGE THE RESULTS OF DESIGN DEFINITION

DE-4.1 Obtain agreement on the security aspects of the design.

DE-4.2 Map the trustworthy secure design characteristics to the system elements.

DE-4.3 Record the trustworthy secure design decisions and the rationale.

DE-4.4 Maintain traceability of the security aspects of the system design.

Note: Traceability is maintained between the trustworthy secure design characteristics and the security architectural entities, system element requirements, interface definitions, analysis results, and verification and validation methods or techniques.

DE-4.5 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.5.3 d)].

Related Publications: [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 21827].

3.4.6 System Analysis

The purpose of the *System Analysis* process is to provide a rigorous basis of data and information for technical understanding to aid decision-making and technical assessments across the life cycle.

3.4.6.1 Security Purpose

- To produce a rigorous basis of data and information for the technical understanding of security aspects to aid decision-making and technical assessments across the life cycle

3.4.6.2 Security Outcomes

- Security aspects of system analysis needs are identified.
- Security aspects of system analysis assumptions and results are validated.
- System analysis results provided for all decisions or technical assessment needs include security aspects.
- Enabling systems or services for the security aspects of system analysis are available.
- Traceability of the security aspects of the system analysis results is established.

3.4.6.3 Security Activities and Tasks

SA-1 PREPARE FOR SYSTEM ANALYSIS

SA-1.1 Define the security aspects of the system analysis strategy.

SA-1.2 Identify the security aspects of the problem or question that require system analysis.

Note: The problem or question may not be driven by or have obvious security consideration or aspects.

SA-1.3 Identify the security-relevant stakeholders of the system analysis.

SA-1.4 Define the scope, objectives, level of fidelity, level of rigor, and level of assurance for the security aspects of system analysis.

SA-1.5 Select the methods to address the security aspects of system analysis.

SA-1.6 Identify the security aspects for enabling systems or services needed to support system analysis.

SA-1.7 Identify and plan for enabling systems or services needed to support the security aspects of system analysis.

SA-1.8 Obtain or acquire access to the security aspects of enabling systems or services to be used in system analysis.

SA-1.9 Identify and validate security-relevant assumptions.

Note 1: This includes assumptions derived from the limits of certainty: what is known, what is insufficiently known, and what is unknown.

Note 2: Assumptions that cannot be validated represent uncertainty and potential risk.

SA-1.10 Plan for and collect the data and inputs needed for the security aspects of the analysis.

References: [ISO 15288, Section 6.4.6.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

SA-2 PERFORM SYSTEM ANALYSIS

SA-2.1 Apply the selected analysis methods to perform the required security-relevant aspects of system analysis.

SA-2.2 Review analysis results for security-relevant quality and validity.

Note: The results are coordinated with associated and previously completed security-relevant analyses. Trustworthiness of the results is determined with the review.

SA-2.3 Establish conclusions and recommendations for the security aspects of the system analysis.

Note: Subject-matter experts are consulted and participate in the formulation of conclusions and recommendations.

SA-2.4 Record the results of the security aspects of the system analysis.

References: [ISO 15288, Section 6.4.6.3 b)].

Related Publications: [ISO 12207, Section 7.1.2.3.1]; [ISO 27034-1]; [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 21827].

SA-3 MANAGE SYSTEM ANALYSIS

SA-3.1 Maintain traceability of the security aspects of the system analysis results.

Note: Bidirectional traceability captures the relationship between the security aspects of the system analysis results, the methods employed, the data used for the analysis, the assumptions, and the context that defines the problem or question addressed.

SA-3.2 Provide the security-relevant artifacts that have been selected for baselines.

Note: This includes general artifacts and security-specific artifacts.

References: [ISO 15288, Section 6.4.6.3 c)].

Related Publications: [ISO 15408-1]; [ISO 15408-2]; [ISO 15408-3]; [ISO 21827].

3.4.7 Implementation

The purpose of the *Implementation* process is to realize a specified system element.

3.4.7.1 Security Purpose

- To transform system security requirements, architecture, and design (including interfaces) into actions that create a trustworthy secure system element according to the practices of the selected implementation technology using appropriate security and non-security technical specialties or disciplines

3.4.7.2 Security Outcomes

- Security-relevant implementation constraints that influence the requirements, architecture, or design are identified.
- A trustworthy secure system element is realized.
- System elements are securely packaged and stored.
- Enabling systems or services for the security aspects of implementation are available.
- Traceability of the security aspects of the implemented system elements is established.

3.4.7.3 Security Activities and Tasks

IP-1 PREPARE FOR IMPLEMENTATION

IP-1.1 Define the trustworthy secure aspects of the implementation strategy.

Note 1: These aspects apply to all system elements that are acquired new, built new, or reused (with or without modification). If the strategy is reuse, then the project needs to determine the extent, source, suitability, and trustworthiness for the purpose of the reused system elements. The implementation strategy includes procedures, fabrication processes, tools and equipment, tolerances, and verification uncertainties, which may introduce weaknesses and vulnerabilities. In the case of repeated system element implementation (e.g., mass production, replacement system elements), the procedures and fabrication processes are defined to achieve consistent and repeatable trustworthy producibility.

Note 2: The security aspects are informed by the targeted level of assurance, security verification uncertainties, and security concerns associated with implementation-related logistics, supply, and distribution of components.

IP-1.2 Identify security-relevant constraints and objectives from implementation in the system security requirements, architecture and design characteristics, or implementation techniques.

IP-1.3 Identify the security aspects for enabling systems, services, and materials needed to support implementation.

IP-1.4 Identify and plan for enabling systems, services, and materials needed to support the security aspects of implementation.

IP-1.5 Obtain or acquire access to the security aspects of enabling systems, services, and materials to be used in implementation.

References: [ISO 15288, Section 6.4.7.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: None.

IP-2 PERFORM IMPLEMENTATION

IP-2.1 Realize or adapt system elements in accordance with the security aspects of the implementation strategy and implementation procedures, as well as security-relevant constraints.

Note: System elements can include:

- *Hardware and Software:* Hardware and software elements are either acquired or fabricated.

  Custom hardware fabrication and software development enable insight into the details of design and implementation. These insights often translate to increased assurance.

  Acquired hardware and software elements may not provide the opportunity to achieve the same insight into design and implementation and may offer more functionality and capability than required. The limits of what can be known about the internals of the elements translate to a level of uncertainty about vulnerability and to the maximum assurance that can be achieved.
- *Firmware:* Firmware exhibits properties of hardware and software. Firmware elements may be acquired or may be developed to realize the software aspects and then fabricated to realize the physical form of the hardware aspects. Firmware elements, therefore, adhere to the security implementation considerations of both hardware and software elements.
- *Services:* System elements implemented by obtaining or leasing services are subject to the same criteria used to acquire hardware, firmware, and software but must also address security considerations associated with utilization and support resources.
- *Utilization and Support Resources:* The security considerations of services acquired or leased must account for the specific roles and responsibilities of individuals of the service/lease

provider and their ability to account for all of the security requirements and constraints associated with the delivery, utilization, and sustainment of the service or capability being leased.

IP-2.2 Place the system element in a secure state for future use, as needed.

Note: This includes protection of the element while stored and in transit, as well as the packaging and labeling of the element.

IP-2.3 Record objective evidence that system elements meet the system security requirements.

References: [ISO 15288, Section 6.4.7.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 7.1.5.3.1]; [ISO 27034-1].

IP-3 MANAGE RESULTS OF IMPLEMENTATION

IP-3.1 Record the security aspects of implementation results and any anomalies encountered.

IP-3.2 Maintain traceability of the security aspects of implemented system elements.

Note: Bidirectional traceability of the security aspects of the implemented system elements to the system security requirements, the security views of the architecture, the security design, and the security interface requirements is maintained.

IP-3.3 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.7.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: None.

3.4.8 Integration

The purpose of the *Integration* process is to synthesize a set of system elements into a realized system that satisfies the system requirements.

3.4.8.1 Security Purpose

- To synthesize a set of system elements into a realized trustworthy secure system that satisfies the system requirements

3.4.8.2 Security Outcomes

- Security-relevant integration constraints that influence requirements, architecture, design, or interfaces and interactions are identified.
- Approaches and checkpoints for the correct secure activation of the identified interfaces and system functions to an initial or established secure state are developed.
- Enabling systems or services for the security aspects of integration are available.
- A trustworthy secure system composed of implemented system elements is integrated.
- Security aspects of system external interfaces (system to external environment) and system internal interfaces (between implemented system elements) are checked.

- Security aspects of integration results and anomalies are identified.
- Traceability of the security aspects of the integrated system elements is established.

3.4.8.3 Security Activities and Tasks

IN-1 PREPARE FOR INTEGRATION

IN-1.1 Identify and define checkpoints for the correct secure activation and integrity of the interfaces and the selected system functions as the system elements are synthesized.

IN-1.2 Define the security aspects of the integration strategy.

Note: Integration is performed to achieve trustworthy secure results using aspects such as secure assembly sequences and checkpoints for the system elements based on established priorities while minimizing integration time and cost and providing appropriate risk treatments.

IN-1.3 Identify the security-relevant constraints and objectives from integration to be incorporated in the system requirements, architecture, or design.

IN-1.4 Identify the security aspects for enabling systems, services, and materials needed to support to support integration.

IN-1.5 Identify and plan for enabling systems, services, and materials needed to support the security aspects of integration.

IN-1.6 Obtain or acquire access to the security aspects of enabling systems, services, and materials to be used in integration.

References: [ISO 15288, Section 6.4.8.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 21827].

IN-2 PERFORM INTEGRATION

IN-2.1 Check interface availability and conformance of the interfaces in accordance with the security aspects of interface definitions and integration schedules.

IN-2.2 Perform actions to address any security-related conformance or availability issues.

IN-2.3 Securely combine the implemented system elements in accordance with planned sequences.

IN-2.4 Securely integrate system element configurations until the complete system is securely synthesized.

IN-2.5 Check for the expected results of interfaces, interconnections, selected functions, and security characteristics.

References: [ISO 15288, Section 6.4.8.3 b)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Sections 6.4.5.3.2, 7.1.6.3.1]; [ISO 27034-1]; [ISO 21827].

IN-3 MANAGE RESULTS OF INTEGRATION

IN-3.1 Record the security aspects of integration results and any anomalies encountered.

Note: Anomaly analyses determine corrective actions that possibly affect the protection capability of the system and the level of assurance that can be obtained.

IN-3.2 Maintain traceability of the security aspects of integrated system elements.

Note: Bidirectional traceability of the security aspects of the integrated system elements to the system security requirements, security views of the architecture, security design, and security interface requirements is maintained. Traceability provides evidence that supports assurance and trustworthiness claims.

IN-3.3 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.8.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

3.4.9 Verification

The purpose of the *Verification* process is to provide objective evidence that a system, system element, or artifact fulfills its specified requirements and characteristics.

3.4.9.1 Security Purpose

- To provide objective evidence that a system, system element, or artifact (e.g., system requirements, architecture description, or design description) fulfills its specified security requirements and characteristics
- To identify security-relevant anomalies[64] in any artifact, implemented system elements, or life cycle processes, and provide the necessary information to determine the resolution of such anomalies

3.4.9.2 Security Outcomes

- Security-relevant verification constraints that influence requirements, architecture, or design are identified.
- Enabling systems or services for the security aspects of verification are available.
- Security aspects of the system, system element, or artifact are verified.
- Security-relevant data that provides information for corrective actions is reported.
- Objective evidence that the realized system fulfills the security requirements and security aspects of the architecture and design is provided.
- Security aspects of verification results and anomalies are identified.
- Traceability of the security aspects of the verified system elements is established.

3.4.9.3 Security Activities and Tasks

VE-1 PREPARE FOR VERIFICATION

VE-1.1 Identify the security aspects within the verification scope and corresponding security verification actions.

Note: Scope includes system, system elements, information items or artifacts that will be verified against applicable requirements, security characteristics, or other security properties. Each

[64] Anomalies include behaviors and outcomes observed but not specified.

verification action description includes what will be verified (e.g., actual system, model, mock-up, prototype, procedure, plan, or other document), the verification method (including any adversity emulation), and the expected result as defined by the success criteria. The security criteria may reflect considerations of strength of function/mechanism, resistance to tamper, misuse or abuse, penetration resistance, level of assurance, absence of flaws, weaknesses, and the absence of unspecified behavior and outcomes.

VE-1.2 Identify the constraints that can potentially limit the feasibility of the security-focused verification actions.

Note: Constraints include technical feasibility; the availability of qualified personnel and verification enablers; the availability of sufficient, relevant, and credible threat data; technology employed (including adversity emulation); the size and complexity of the system element or artifact; and the cost and time allotted for the verification.

VE-1.3 Select appropriate security verification methods and the associated success criteria for each security verification action.

Note: The methods and techniques are selected to provide the evidence required to achieve the expected results with the desired level of assurance.

VE-1.4 Define the security aspects of the verification strategy.

Note: This includes the approach used to incorporate security considerations into all verification actions, considering trade-offs between scope, depth, and rigor needed for the desired level of assurance and the given constraints.

VE-1.5 Identify the security-relevant constraints and objectives that result from the security aspects of the verification strategy to be incorporated into the system requirements, architecture, and design.

VE-1.6 Identify the security aspects for enabling systems or services needed to support verification.

VE-1.7 Identify and plan for enabling systems or services needed to support the security aspects of verification.

VE-1.8 Obtain or acquire access to the security aspects of enabling systems or services to be used in verification.

References: [ISO 15288, Section 6.4.9.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4] [ISO 29119-1]; [ISO 29119-2]; [ISO 29119-3]; [ISO 29119-4]; [ISO 29148].

Related Publications: [ISO 12207, Section 7.2.4.3.1]; [ISO 21827].

VE-2 PERFORM VERIFICATION

VE-2.1 Define the security aspects of the verification procedures, each supporting one or a set of verification actions.

Note: The procedures identify the security purpose of verification, the success criteria (expected results), the verification method to be applied, the necessary enabling systems (e.g., facilities, equipment, etc.), and the environmental conditions to perform each verification procedure (e.g., resources, qualified personnel, adversity emulations, etc.).

VE-2.2 Perform security verification procedures.

References: [ISO 15288, Section 6.4.9.3 b)].

Related Publications: [ISO 12207, Sections 6.4.6.3.1, 7.1.7.3.1, 7.2.4.3.2]; [ISO 27034-1]; [ISO 21827].

VE-3 MANAGE RESULTS OF VERIFICATION

VE-3.1 Record the security aspects of verification results and any anomalies encountered.

VE-3.2 Obtain agreement from the approval authority that the system, system element, or artifact meets the specified system security requirements.

Note: There may be multiple approval authorities with security-related responsibilities.

VE-3.3 Maintain traceability of the security aspects of verification.

Note: Bidirectional traceability is maintained between the verified security aspects of system elements and the system security requirements, architecture, design, and interface requirements. This traceability includes verification results or evidence, such as security-relevant anomalies, deviations, or requirement satisfaction.

VE-3.4 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.9.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27034-1].

Related Publications: [ISO 21827].

3.4.10 Transition

The purpose of the *Transition* process is to establish a capability for a system to provide services specified by stakeholder requirements in the operational environment.

3.4.10.1 Security Purpose

- To preserve the system's verified security characteristics during the orderly and planned transition of the system to be operable in the intended environment, which may be a new or changed environment

3.4.10.2 Security Outcomes

- Security-relevant transition constraints that influence system requirements, architecture, or design are identified.
- Enabling systems or services for the security aspects of transition are available.
- The prepared site satisfies security criteria.
- The system is installed in its operational environment and is capable of delivering its specified functions in a trustworthy secure manner.
- Operators, users, and other stakeholders necessary to the system utilization and support are trained in the system's security capabilities, mechanisms, and features.
- Security-relevant transition results and anomalies are identified.
- The installed system is activated and ready for trustworthy secure operation.
- Traceability of the security aspects of the transitioned elements is established.

3.4.10.3 Security Activities and Tasks

TR-1 PREPARE FOR TRANSITION

TR-1.1 Define the security aspects of the transition strategy.

Note: The transition strategy includes all security-relevant activities, from site delivery and installation through deployment and commissioning of the system, as well as all security-relevant stakeholders, including human operators. The strategy also includes security roles and responsibilities, facilities security considerations, secure shipping and receiving, contingency back out plans, security training, security aspects of installation acceptance demonstration tasks, secure operational readiness reviews, secure operations commencement, transition security success criteria, rights of secure access, data rights, and integration with other plans. System commissioning is considered along with the secure decommissioning of the old system when one exists. In this case, the Transition and Disposal processes are used concurrently.

TR-1.2 Identify and define any security-relevant facility or site changes needed.

TR-1.3 Identify the security-relevant constraints and objectives from the security aspects of transition to be incorporated into the system requirements, architecture, and design.

TR-1.4 Identify and arrange the security training of operators, users, and other stakeholders necessary to the system utilization and support.

TR-1.5 Identify the security aspects for enabling systems or services needed to support transition.

TR-1.6 Identify and plan for enabling systems or services needed to support the security aspects of transition.

TR-1.7 Obtain or acquire access to the security aspects of enabling systems or services to be used in transition.

TR-1.8 Identify security aspects, and arrange for the secure shipping and receiving of system elements and enabling systems.

References: [ISO 15288, Section 6.4.10.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: None.

TR-2 PERFORM TRANSITION

TR-2.1 Prepare the site of operation in accordance with secure installation requirements.

TR-2.2 Securely deliver the system for installation at the correct location and time.

Note: Secure delivery considers the various forms, means, and methods that accomplish end-to-end transport of system elements to ensure that system elements are not tampered with during transport. Items and packages are delivered to the intended recipient and only to the intended recipient, which may mean shipping with more lead time to account for additional security.

TR-2.3 Install the system in its operational environment in accordance with the secure installation strategy, and establish secure interconnections to its environment.

TR-2.4 Demonstrate trustworthy secure system installation.

Note: The installation and connection procedures are to be properly verified to provide confidence that the intended system configuration across all system modes and states is achieved. This includes completion of the acceptance tests defined in agreements. These tests include security aspects associated with physical connections between the system and the environment.

TR-2.5 Provide security training for the operators, users, and other stakeholders necessary for system utilization and support.

TR-2.6 Perform security activation and checkout of the system.

Note: Security activation and checkout shows that the system can initialize to its initial secure operational state for all defined modes of operation and accounts for all interconnections to other systems across physical, virtual, and wireless interfaces.

TR-2.7 Demonstrate that the installed system is capable of delivering its required functions in a trustworthy secure manner.

TR-2.8 Demonstrate that the security functions provided by the system and the effects of the security functions are sustainable by enabling systems.

TR-2.9 Review the security trustworthiness of the system for operational readiness.

Note: The results of installation, operational, and enabling system checkouts are reviewed to determine if the security performance and effectiveness are sufficient to justify operational use.

TR-2.10 Commission the system for secure operation.

Note: This includes providing security support to users and operators at the time of the system commissioning.

References: [ISO 15288, Section 6.4.10.3 b)].

Related Publications: [ISO 12207, Sections 6.4.7.3.1, 6.4.8.3.1, 6.4.9.3.2].

TR-3 MANAGE RESULTS OF TRANSITION

TR-3.1 Record the security aspects of transition results and any anomalies encountered.

TR-3.2 Record the security aspects of operational incidents and problems, and track their resolution.

TR-3.3 Maintain traceability of the security aspects of transitioned system elements.

Note: Bidirectional traceability is maintained between all identified security aspects and supporting data associated with the transition strategy and the system requirements, system architecture, and system design.

TR-3.4 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.10.3 c)].

Related Publications: None.

3.4.11 Validation

The purpose of the *Validation* process is to provide objective evidence that the system, when in use, fulfills its business or mission objectives and stakeholder requirements, achieving its intended use in its intended operational environment.

3.4.11.1 Security Purpose

- To provide objective evidence that the system, when in use, fulfills the protection needs associated with its business or mission objectives and the stakeholder requirements, achieving its intended use in its intended operational environment in a trustworthy secure manner

3.4.11.2 Security Outcomes

- Security validation criteria are defined.
- The availability of security services required by stakeholders is confirmed.
- Security-relevant validation constraints that influence system requirements, architecture, or design are identified.
- Security aspects of the system, system element, or artifact are validated.
- Enabling systems or services for the security aspects of validation are available.
- Security-focused validation results and anomalies are identified.
- Objective evidence of the successful validation of security aspects is provided.
- Traceability of the validated security aspects of the system, system elements, and artifacts is established.

3.4.11.3 Security Activities and Tasks

VA-1 PREPARE FOR VALIDATION

VA-1.1 Identify the security aspects within the validation scope and corresponding security validation actions.

Note: The security aspects of validation focus on the stakeholders' protection needs, concerns, and associated stakeholder security requirements. The scope includes system elements, the entire system, or any artifact that impacts the stakeholder's confidence in the system and the decision to accept the system as being trustworthy for its intended use.

VA-1.2 Identify the constraints that can potentially limit the feasibility of the security validation actions.

Note: Constraints may include the level of assurance and the availability of business or mission stakeholders to support validation activities; the availability of sufficient, relevant, and credible threat data; the limits on conducting validation activities in actual operational conditions across all business and mission modes and associated system states and modes; technology employed; the size and complexity of the system element or artifact; and the cost and time allotted for validation activities.

VA-1.3 Select appropriate security validation methods and the associated success criteria for each security validation action.

Note: Adversity emulation, including penetration testing and emulating abuse and misuse, is included.

VA-1.4 Develop the security aspects of the validation strategy.

Note: The security aspects of the validation strategy address the approach to incorporate security considerations into all validation actions, considering trade-offs between scope, depth, and rigor needed for the desired level of assurance and the given constraints.

VA-1.5 Identify the security-relevant system constraints that result from the security aspects of the validation strategy to be incorporated in the stakeholder protection needs and the requirements transformed from those needs.

Note: These constraints are associated with the clarity and accuracy of the expression of needs and requirements in order to achieve the desired level of assurance with certainty and repeatability.

VA-1.6 Identify the security aspects for enabling systems or services needed to support validation.

VA-1.7 Identify and plan for enabling systems or services to support the security aspects of validation.

VA-1.8 Obtain or acquire access to the security aspects of enabling systems or services to be used to support validation.

References: [ISO 15288, Section 6.4.11.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 7.2.5.3.1]; [ISO 21827].

VA-2 PERFORM VALIDATION

VA-2.1 Define the security aspects of the validation procedures, each supporting one or a set of validation actions.

Note: This includes the identification of the validation methods or techniques to be employed, the qualifications of individuals conducting the validation, and any specialized equipment that may be needed, such as what may be required to emulate environmental adversities.

VA-2.2 Perform security validation procedures.

Note 1: Security-focused validation actions from the execution of validation procedures contribute to demonstrating that the system is sufficiently trustworthy.

Note 2: The performance of a security-focused validation action consists of capturing a result from the execution of the procedure, comparing the obtained result with the expected result, deducing the degree of compliance of the element, and deciding about the acceptability of compliance if uncertainty remains.

References: [ISO 15288, Section 6.4.11.3 b)].

Related Publications: [ISO 12207, Sections 6.4.8.3.1, 7.2.5.3.2]; [ISO 21827].

VA-3 MANAGE RESULTS OF VALIDATION

VA-3.1 Record the security aspects of validation results and any anomalies encountered.

Note: The recorded validation results include nonconformance issues, anomalies, or problems that are potentially security related. These results inform the analyses to determine causes and enable corrective or improvement actions. Corrective actions may affect the security aspects of the system architecture definition, design definition, system security requirements and associated constraints, the level of assurance that can be obtained, and/or the implementation strategy, including its security aspects.

VA-3.2 Record the security characteristics of operational incidents and problems, and track their resolution.

Note: Incidents that occur in the operational environment of the system are recorded and subsequently correlated to validation activities and results. This is an important feedback loop for continuous improvement in the engineering of trustworthy secure systems.

VA-3.3 Obtain agreement that security validation criteria have been met.

VA-3.4 Maintain traceability of the security aspects of validation.

Note: Bidirectional traceability of the security aspects of validated system elements to stakeholder protection needs, security concerns, and security requirements is maintained. Traceability demonstrates completeness of the validation process and provides evidence that supports assurance and trustworthiness claims.

VA-3.5 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.11.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

3.4.12 Operation

The purpose of the *Operation* process is to use the system to deliver its services.

3.4.12.1 Security Purpose

- To inform the security aspects of the requirements and constraints to securely operate the system and monitor the security aspects of products, services, and operator-system performance
- To identify and analyze security-relevant operational anomalies

3.4.12.2 Security Outcomes

- Security aspects of operation constraints that influence system requirements, architecture, or design are identified.
- Enabling systems, services, and material for the security aspects of operation are available.
- Trained and qualified personnel who can securely operate the system are available.
- System products or services that meet stakeholder security requirements are delivered.
- Security aspects of system performance during operation are monitored.
- Security support to stakeholders is provided.

3.4.12.3 Security Activities and Tasks

OP-1 PREPARE FOR OPERATION

OP-1.1 Define the security aspects of the operation strategy.

Note 1: This includes the approach to enable the continuous secure operation and use of the system and its security services, as well as the provision of support to operations elements to address anomalies identified during operation and use of the system. It also includes:

- The capacity, availability, schedule considerations, and security of products or services as they are introduced, routinely operated, and disposed (including contingency operations)
- The human resources strategy and security qualification requirements for personnel including all associated security-related training and personnel compliance requirements
- The security aspects of release and re-acceptance criteria and schedules of the system to permit modifications that sustain the security aspects of existing or enhanced products or services

- The approach to implement the operational modes in the System Operational Concept, including normal and contingency operations
- The secure approaches for contingency, degraded, alternative, training, and other modes of operation, as well as transition within and between modes while considering resilience in the face of adversity
- Measures for operation that will provide security insights into performance levels
- The approach to achieve situational awareness to determine security-relevant consequences

Note 2: This includes planning for securely starting the system, halting the system, shutting down the system, operating the system in a training mode, secure implementation of work-around procedures to restore operation, performing back-out and restore operations, operating in any degraded mode, or alternative modes for special conditions. If needed, the operator performs the necessary steps to enter into contingency operations and possibly power down the system. Contingency operations are performed in accordance with pre-established procedures for such an event.

Note 3: There may be a need to plan for certain modes of operation for which security functions and services are reduced or eliminated to achieve more critical system functions and services or to carry out certain maintenance or periodic testing. Predetermined procedures for entering and exiting such modes would be followed.

OP-1.2 Identify the constraints and objectives that result from the security aspects of operation to be incorporated into the system requirements, architecture, and design.

OP-1.3 Identify the security aspects for enabling systems and services needed to support operation.

OP-1.4 Identify and plan for enabling systems or services needed to support the security aspects of operation.

OP-1.5 Obtain or acquire access to the security aspects of enabling systems or services to be used in operation.

OP-1.6 Identify or define security training and qualification requirements to sustain the workforce needed for secure system operation.

Note: Security qualification and training includes role and function-oriented competency, proficiency, certification, and other criteria to securely operate and use the system in all of its defined modes or states.

OP-1.7 Assign trained and qualified personnel needed for secure system operation.

References: [ISO 15288, Section 6.4.12.3 a)]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.4.9.3.1]; [ISO 21827].

OP-2 PERFORM OPERATION

OP-2.1 Securely use the system in its intended operational environment.

OP-2.2 Apply materials and other resources as required to securely operate the system and sustain its product and service capabilities.

Note 1: Materials and resources are provided by logistical actions. Logistics is discussed as part of the maintenance process.

Note 2: Operational personnel may perform system modification and support activities, such as software updates.

OP-2.3 Monitor system operations for deviations from intended behavior and outcomes.

Note: This includes managing adherence to the operation strategy and operational procedures (the operations conducted by personnel) and monitoring that the system is operated in a secure manner and compliant with regulations, procedures, and directives. This also includes monitoring for anomalies that may not be directly observable as system behavior and may or may not be obviously security relevant.

OP-2.4 Use the measures defined in the strategy, and analyze them to confirm that system security performance is within acceptable parameters.

Note: System monitoring includes reviewing whether the performance is within established security-relevant thresholds, periodic instrument readings are acceptable, and service and response times are acceptable. Operator feedback and suggestions are useful input for improving the security aspects of system operational performance.

OP-2.5 Identify and record when system security or service performance is not within acceptable parameters.

References: [ISO 15288, Section 6.4.12.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Section 6.4.9.3.3]; [ISO 21827].

OP-3 MANAGE RESULTS OF OPERATION

OP-3.1 Record the results of secure operations and any anomalies encountered.

Note: Anomalies include those associated with the operation strategy, the operation of enabling systems, the execution of the operation, and incorrect system definition, all of which may be due to security issues or may result in security issues.

OP-3.2 Record the security aspects of operational incidents and problems, and track their resolution.

OP-3.3 Maintain traceability of the security aspects of the operation elements.

OP-3.4 Provide the security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.12.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

OP-4 SUPPORT STAKEHOLDERS

OP-4.1 Provide security assistance and consultation to stakeholders as requested.

Note: Assistance and consultation includes the provision or recommendation of sources for security-relevant training, security aspects of documentation, vulnerability resolution, security reporting (including cyber security), and other security-relevant support services that enable effective and secure use of the product or service.

OP-4.2 Record and monitor requests and subsequent actions for security support.

OP-4.3 Determine the degree to which the security aspects of delivered products and services satisfy the needs of stakeholders.

References: [ISO 15288, Section 6.4.12.3 d)].

Related Publications: [ISO 12207, Sections 6.4.9.3.4, 6.4.9.3.5]; [ISO 21827].

3.4.13 Maintenance

The purpose of the *Maintenance* process is to sustain the capability of the system to provide a product or service.

[ISO 15288] *Reprinted with permission from IEEE, Copyright IEEE 2015, All rights reserved.*

3.4.13.1 Security Purpose

- To establish the security aspects of requirements and constraints to securely sustain the capability of the system to provide a product or service

Note: Secure sustainment includes all maintenance and logistics activities for the packaging, handling, storage, and transportation of replacement system elements.

3.4.13.2 Security Outcomes

- Security aspects of maintenance and logistics constraints that influence system requirements, architecture, or design are identified.
- Enabling systems or services needed for the security aspects of system maintenance and logistics are available.
- Replacement, repaired, or modified system elements are securely made available.
- The need for required security-relevant maintenance and logistics actions is reported.
- Security-relevant failures and life cycle data, including associated costs, are determined.

3.4.13.3 Security Activities and Tasks

MA-1 PREPARE FOR MAINTENANCE AND LOGISTICS

MA-1.1 Define the security aspects of the maintenance strategy.

Note: The maintenance strategy seeks to preserve the secure capability and performance of the delivered system. The security aspects of the maintenance strategy generally include:

- The secure transition of the system and system elements into a secure maintenance mode or state, as well as the secure transition back to operation.
- An approach to help ensure that sourced materials and system elements that do not meet specified quality, origin, and functionality (e.g., counterfeit) are not introduced into the system.
- The skill and personnel levels required to effect repairs, replacements, and restoration accounting for maintenance staff requirements and any relevant legislation regarding health, safety, security, and the environment.
- Maintenance measures that provide insight into the security aspects of performance levels, effectiveness, and efficiency.

MA-1.2 Define the security aspects of the logistics strategy.

Note: The logistics strategy defines the specific security considerations required to perform logistics throughout the life cycle. This generally includes:

- Acquisition logistics to help ensure that security implications are considered early during the development stage.
- Operations logistics to help ensure that the necessary material and resources, in the right quantity and quality, are securely made available at the right place and time throughout the

utilization and support stages; considerations for securely making material and resources available include identification and marking, packaging, distribution, handling, and provisioning.

- The security criteria for storage locations and conditions, as well as the number and type of replacement system security-specific elements, their anticipated replacement rate, and their storage life and renewal frequency.

MA-1.3 Identify the security-relevant constraints and objectives that result from the security aspects of maintenance and logistics to be incorporated into the system requirements, architecture, and design.

MA-1.4 Identify trade-offs such that the security aspects of the system and associated maintenance and logistics actions result in a solution that is trustworthy, secure, affordable, operable, supportable, and sustainable.

Note: The cost of secure maintenance and logistics should be considered within the lifetime cost of the system.

MA-1.5 Identify the security aspects for enabling systems, products, and services needed to support maintenance and logistics.

MA-1.6 Identify and plan for enabling systems, products, and services needed to support the security aspects of maintenance and logistics.

MA-1.7 Obtain or acquire access to the security aspects of enabling systems, products, and services to be used in maintenance and logistics.

References: [ISO 15288, Section 6.4.13.3 a)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 12207, Section 6.4.10.3.1]; [ISO 21827].

MA-2 PERFORM MAINTENANCE

Note: The need to perform maintenance may be driven by the need to address explicit security issues, incidents, or failures. All maintenance actions must be accomplished in a secure manner with the understanding that some actions may have a direct effect on the security posture of the system.

MA-2.1 Monitor and review stakeholder requirements and incident and problem reports to identify security-relevant corrective, preventive, adaptive, additive, or perfective maintenance needs.

Note: Security-relevant maintenance needs include those needs that are direct (e.g., an identified security incident) or indirect (e.g., considerations to securely address a maintenance need).

MA-2.2 Record the security aspects of maintenance incidents and problems, and track their secure resolution.

MA-2.3 Analyze the impact of changes introduced by maintenance actions on the security aspects of the system and system elements.

MA-2.4 Upon encountering faults that cause a system failure, securely restore the system to secure operational status.

Note: Secure restoration means that the maintenance action itself does not worsen the secure state or condition of the system.

MA-2.5 Securely correct anomalies (e.g., defects, errors, and faults), and replace or upgrade system elements.

MA-2.6 Perform preventive maintenance by securely replacing or servicing system elements prior to failure.

MA-2.7 Securely perform adaptive, additive, or perfective maintenance as required.

References: [ISO 15288, Section 6.4.13.3 b)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 12207, Sections 6.4.10.3.2, 6.4.10.3.3, 6.4.10.3.4, 6.4.10.3.5]; [ISO 21827].

MA-3 PERFORM LOGISTICS SUPPORT

MA-3.1 Perform the security aspects of acquisition logistics.

MA-3.2 Perform the security aspects of operational logistics.

MA-3.3 Implement mechanisms for the secure logistics needed during the life cycle.

Note 1: These mechanisms enable secure packaging, handling, storage, and transportation.

Note 2: These mechanisms aid in the prevention and detection of counterfeits, tampering, substitution, and redirection.

MA-3.4 Confirm that the security aspects of logistics actions are implemented.

Note: The security aspects of logistics actions satisfy both logistics protection concerns and the need to meet repair rates, replenishment levels, and planned schedules.

References: [ISO 15288, Section 6.4.13.3 c)]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4]; [ISO 27036-1]; [ISO 27036-2]; [ISO 27036-3].

Related Publications: [ISO 21827].

MA-4 MANAGE RESULTS OF MAINTENANCE AND LOGISTICS

MA-4.1 Record the security aspects of maintenance and logistics results and any anomalies encountered.

MA-4.2 Record maintenance and logistics security incidents and problems, and track their secure resolution.

MA-4.3 Identify and record the security-relevant trends of incidents, problems, and maintenance and logistics actions.

MA-4.4 Maintain traceability of the security aspects of maintenance and logistics.

MA-4.5 Provide security-relevant artifacts that have been selected for baselines.

MA-4.6 Monitor customer satisfaction with the security aspects of the system, maintenance, and logistics.

References: [ISO 15288, Section 6.4.13.3 d)]; [ISO 10004]; [ISO 15026-1]; [ISO 15026-2]; [ISO 15026-3]; [ISO 15026-4].

Related Publications: [ISO 21827].

3.4.14 Disposal

The purpose of the *Disposal* process is to end the existence of a system element or system for a specified intended use, appropriately handle replaced or retired elements and any waste products, and properly attend to identified critical disposal needs (e.g., per an agreement, per organizational policy, or for environmental, legal, safety, or security aspects).

[ISO 15288] *Reprinted with permission from IEEE, Copyright IEEE 2015, All rights reserved.*

3.4.14.1 Security Purpose

- To provide the aspects needed to securely end the existence of a system element or system for a specified intended use and securely preserve or destroy the associated data and information

3.4.14.2 Security Outcomes

- Secure disposal constraints that influence system requirements, architecture, design, and implementation are identified.
- Enabling systems or services for the security aspects of disposal are available.
- System elements or waste products are destroyed, stored, reclaimed, or recycled in accordance with safety and security requirements.
- The environment is securely returned to its original secure or an agreed-upon secure state.
- Records of the security aspects of disposal actions and analysis are available.

3.4.14.3 Security Activities and Tasks

DS-1 PREPARE FOR DISPOSAL

DS-1.1 Define the security aspects of the disposal strategy.

Note: The security aspects address securely terminating system functions and services, transforming the system and environment into an acceptable secure state, addressing security concerns, and transitioning the system and system elements for future use. The disposal strategy determines approaches, schedules, resources, specific considerations of secure disposal, and the effectiveness and completeness of secure disposal and disposition actions.

- *Permanent termination of system functions and delivery of services:* The security aspects address the removal, decommissioning, or destruction of the associated system elements while preserving the security posture of any remaining functions and services.
- *Transform the system and environment into an acceptable state:* The security aspects address any alterations made to the system, its operation, and the environment to ensure that stakeholder protection needs and concerns are addressed by the remaining portions of the system and the functions and services it provides. When the entire system is removed, the security aspects address alterations to the environment to return it to its original or agreed-upon secure state.
- *Address security concerns for material, data, and information:* The security aspects address protections for sensitive components, technology, information, and data removed from service, dismantled, stored, prepared for reuse, or destroyed. The aspects may include the duration of protection level/state, downgrades, releasability, and criteria that define authorized access and use during the storage period. The protection needs for disposal are defined by stakeholders and agreements and may be subject to regulatory requirements, expectations, and constraints.
- *Transition the system and system elements for future use:* The security aspects address the transition of the system or system elements for future use in a modified or adapted form, including legacy migration and return to service. The security aspects may include constraints, limitations, or other criteria to enable recovery of the systems' functions and services within a specified time period or to ensure security-oriented interoperability with

future enabling systems and other systems. These aspects may also include periodic inspections to account for the security posture and return-to-service readiness of stored system elements, associated data and information, and all supporting operations and sustainment support materials. The security aspects apply to all system functions and services and are not limited to only security protection-oriented functions and services of the system.

DS-1.2 Identify the security-relevant constraints and objectives of disposal on the system requirements, architecture and design characteristics, and implementation techniques.

DS-1.3 Identify the security aspects for enabling systems or services needed to support disposal.

DS-1.4 Identify and plan for enabling systems or services needed to support the security aspects of disposal.

DS-1.5 Obtain or acquire access to the security aspects of enabling systems or services to be used in disposal.

DS-1.6 Specify security criteria for containment facilities, storage locations, inspection, and storage periods (if the system is to be stored).

DS-1.7 Define the security aspects of preventive methods to preclude disposed elements and materials that should not be repurposed, reclaimed, or reused from re-entering the supply chain.

References: [ISO 15288, Section 6.4.14.3 a)].

Related Publications: [ISO 12207, Section 6.4.11.3.1]; [ISO 21827].

DS-2 PERFORM DISPOSAL

DS-2.1 Securely deactivate the system or system element to prepare it for secure removal from operation.

Note: Deactivation is accomplished to preserve the security posture of the system.

DS-2.2 Securely remove the system, system element, or waste material from use or production for appropriate secure disposition and action.

DS-2.3 Securely withdraw impacted operating staff from the system or system element, and record relevant secure operation knowledge.

DS-2.4 Securely disassemble the system or system element into manageable elements to facilitate its secure removal for reuse, recycling, reconditioning, overhaul, archiving, or destruction.

Note: Secure disassembly preserves the security characteristics of the system elements that are not removed.

DS-2.5 Securely handle system elements and their parts that are not intended for reuse in a manner that will help ensure that they do not get back into the supply chain.

DS-2.6 Conduct secure sanitization and destruction of the system elements and life cycle artifacts.

Note 1: Governing agreements, laws, and regulations determine the appropriate means to sanitize and destroy data, information, and systems elements that contain data and information, as well as retention periods before sanitization and destruction can occur.

Note 2: Sanitization and destruction techniques include clearing, purging, cryptographic erase, physical modification, and physical destruction.

Note 3: Sanitization and destruction techniques and methods may be specific to data, information, and system element type.

References: [ISO 15288, Section 6.4.14.3 b)].

Related Publications: [ISO 12207, Section 6.4.11.3.2]; [ISO 21827].

DS-3 FINALIZE THE DISPOSAL

DS-3.1 Confirm that no detrimental security factors exist following disposal.

DS-3.2 Return the environment to its original secure state or to a secure state specified by agreement.

DS-3.3 Securely archive data and information gathered through the lifetime of the system to permit audits and reviews in the event of long-term hazards to health, safety, security, and the environment and to permit future system creators and users to securely build a knowledge base from past experiences.

DS-3.4 Provide security-relevant artifacts that have been selected for baselines.

References: [ISO 15288, Section 6.4.14.3 c)].

Related Publications: [ISO 21827].

REFERENCES

KEY REFERENCES RELATED TO SYSTEMS SECURITY ENGINEERING

LAWS AND EXECUTIVE ORDERS	
[EGOV]	E-Government Act [incl. FISMA] (P.L. 107-347), December 2002. https://www.govinfo.gov/app/details/PLAW-107publ347
[EO 14028]	Executive Order 14028 (2021), *Improving the Nation's Cybersecurity.* (The White House, Washington, DC), May 12, 2021. https://www.federalregister.gov/d/2021-10460
[FISMA]	Federal Information Security Modernization Act (P.L. 113-283), December 2014. https://www.govinfo.gov/app/details/PLAW-113publ283
[FOIA96]	Freedom of Information Act (FOIA), 5 U.S.C. § 552, As Amended By Public Law No. 104-231, 110 Stat. 3048, Electronic Freedom of Information Act Amendments of 1996. https://www.govinfo.gov/app/details/PLAW-104publ231
POLICIES, DIRECTIVES, AND INSTRUCTIONS	
[CNSSI 4009]	Committee on National Security Systems Instruction (CNSSI) No. 4009, *Committee on National Security Systems (CNSS) Glossary*, April 2015. https://www.cnss.gov/CNSS/issuances/Instructions.cfm
[DODD 8140.01]	Department of Defense (DoD) Directive 8140.01, *Cyberspace Workforce Management*, October 2020. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodd/814001p.PDF?ver=si7QmZONMCW2tStUt4ws3Q%3D%3D
[OMB A-130]	Office of Management and Budget (2016) Managing Information as a Strategic Resource. (The White House, Washington, DC), OMB Circular A-130, July 2016. https://www.whitehouse.gov/sites/whitehouse.gov/files/omb/circulars/A130/a130revised.pdf
[OMB M-19-03]	Office of Management and Budget (2018) Strengthening the Cybersecurity of Federal Agencies by enhancing the High Value Asset Program. (The White House, Washington, DC), OMB Memorandum M-19-03, December 2018. https://www.whitehouse.gov/wp-content/uploads/2018/12/M-19-03.pdf

STANDARDS AND GUIDELINES

[ANSI G-043B] American National Standards Institute (ANSI)/American Institute of Aeronautics and Astronautics (AIAA) G-043B-2018, *Guide To The Preparation Of Operational Concept Documents*, April/May 2018. https://webstore.ansi.org/Standards/AIAA/ANSIAIAA043B2018

[EIA 649C] Electronic Industries Alliance (EIA) 649C, *Configuration Management Standard*, February 2019. https://www.sae.org/standards/content/eia649c

[GSNCS18] Goal Structuring Notation Community Standard, Version 2, The Assurance Case Working Group, January 2018. https://scsc.uk/r141B:1?t=1

[IEEE 610.12] Institute of Electrical and Electronics Engineers (IEEE) Std. 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology*, December 1990. https://standards.ieee.org/standard/610_12-1990.html

[IEEE 828] Institute of Electrical and Electronics Engineers (IEEE) Std. 828-2012, *IEEE Standard for Configuration Management in Systems and Software Engineering*, IEEE Computer Society, March 2012. https://standards.ieee.org/standard/828-2012.html

[ISO 73] International Organization for Standardization (ISO) Guide 73:2009, *Risk management – Vocabulary*, November 2009. https://www.iso.org/standard/44651.html

[ISO 7498-2] International Organization for Standardization (ISO) ISO 7498-2:1989, *Information processing systems — Open Systems Interconnection — Basic Reference Model — Part 2: Security Architecture*, February 1989. https://www.iso.org/standard/14256.html

[ISO 9000] International Organization for Standardization (ISO) 9000:2015, *Quality management systems – Fundamentals and vocabulary*, September 2015. https://www.iso.org/standard/45481.html

[ISO 9001] International Organization for Standardization (ISO) 9001:2015, *Quality management systems – Requirements*, September 2015. https://www.iso.org/standard/62085.html

[ISO 9241] International Organization for Standardization (ISO) 9241-210:2010, *Ergonomics of human-system interaction — Part 210: Human-centered design for interactive systems*, March 2010. https://www.iso.org/standard/52075.html

[ISO 10004] International Organization for Standardization (ISO) 10004:2018, *Quality management – Customer satisfaction – Guidelines for monitoring and managing*, August 2018. https://www.iso.org/standard/71582.html

[ISO 10007] International Organization for Standardization (ISO) 10007:2003, *Quality management systems – Guidelines for configuration management*, March 2017.
https://www.iso.org/standard/70400.html

[ISO 12207] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 12207:2008, *Systems and software engineering – Software life cycle processes*, November 2017.
https://www.iso.org/standard/63712.html

[ISO 13008] International Organization for Standardization (ISO) 13008:2012, *Information and documentation — Digital records conversion and migration process*, June 2012.
https://www.iso.org/standard/52326.html

[ISO 14258] International Organization for Standardization (ISO) 14258:1998, *Industrial automation systems — Concepts and rules for enterprise models*, September 1998.
https://www.iso.org/standard/24020.html

[ISO 15026-1] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 15026-1:2019, *Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary*, March 2019.
https://www.iso.org/standard/73567.html

[ISO 15026-2] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15026-2:2011, *Systems and software engineering -- Systems and software assurance -- Part 2: Assurance case*, February 2011.
https://www.iso.org/standard/80625.html

[ISO 15026-3] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15026-3:2015, *Systems and software engineering -- Systems and software assurance -- Part 3: System integrity levels*, November 2015.
https://www.iso.org/standard/64842.html

[ISO 15026-4] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15026-4:2012, *Systems and software engineering -- Systems and software assurance -- Part 4: Assurance in the life cycle*, October 2012.
https://www.iso.org/standard/59927.html

[ISO 15288] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 15288:2015, *Systems and software engineering —Systems life cycle processes*, May 2015.
https://www.iso.org/standard/63711.h1ml

[ISO 15408-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15408-1:2009, *Information technology — Security techniques — Evaluation criteria for IT security — Part 1: Introduction and general model*. https://www.iso.org/standard/72891.html

[ISO 15408-2] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15408-2:2008, *Information technology — Security techniques — Evaluation criteria for IT security — Part 2: Security functional requirements*. https://www.iso.org/standard/72892.html

[ISO 15408-3] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15408-3:2008, *Information technology — Security techniques — Evaluation criteria for IT security — Part 3: Security assurance requirements*. https://www.iso.org/standard/46413.html

[ISO 15939] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 15939:2017, *Systems and software engineering — Measurement process*, May 2017. https://www.iso.org/standard/71197.html

[ISO 16085] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 16085:2021, *Systems and software engineering — Life cycle processes — Risk management*, January 2021. https://www.iso.org/standard/74371.html

[ISO 16290] International Organization for Standardization (ISO) 16290:2013, *Space systems — Definition of the Technology Readiness Levels (TRLs) and their criteria of assessment*, November 2013. https://www.iso.org/standard/56064.html

[ISO 18152] International Organization for Standardization/Technical Specification (ISO/TS) 18152:2010, *Ergonomics of human-system interaction — Specification for the process assessment of human-system issues*, June 2010. https://www.iso.org/standard/56174.html

[ISO 18307] International Organization for Standardization/Technical Report (ISO/TR) 18307:2001, *Health informatics — Interoperability and compatibility in messaging and communication standards — Key characteristics*, December 2001. https://www.iso.org/standard/33396.html

[ISO 19014] International Organization for Standardization (ISO) 19014:2020, *Earth-moving machinery — Functional safety — Part 4: Design and evaluation of software and data transmission for safety-related parts of the control system*, July 2020.
https://www.iso.org/standard/70718.html

[ISO 21827] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 21827:2008, *Information technology — Security techniques — Systems Security Engineering — Capability Maturity Model® (SSE-CMM®)*, October 2008.
https://www.iso.org/standard/44716.html

[ISO 21839] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 21839:2019, *Systems and software engineering — System of systems (SoS) considerations in life cycle stages of a system*, July 2019.
https://www.iso.org/standard/71955.html

[ISO 21840] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 21840:2019, *Systems and software engineering — Guidelines for the utilization of ISO/IEC/IEEE 15288 in the context of system of systems (SoS)*, December 2019.
https://www.iso.org/standard/71956.html

[ISO 21841] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 21841:2019, Systems and software engineering — Taxonomy of systems of systems, July 2019.
https://www.iso.org/standard/71957.html

[ISO 24748-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) TR 24748-1:2010, *Systems and software engineering — Life cycle management — Part 1: Guide for life cycle management*, October 2010.
https://www.iso.org/standard/50502.html

[ISO 24748-6] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) TR 24748-6:2016, *Systems and software engineering — Life cycle management — Part 6: System integration engineering*, December 2016.
https://www.iso.org/standard/66433.html

[ISO 24765] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 24765:2017, *Systems and software engineering — Vocabulary*, September 2017.
https://www.iso.org/standard/71952.html

[ISO 24774] International Organization for Standardization/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 24774:2021, *Systems and software engineering — Life cycle management — Specification for process description*, May 2021.
https://www.iso.org/standard/78981.html

[ISO 25010] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 25010:2011, *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models*, March 2011.
https://www.iso.org/standard/35733.html

[ISO 25030] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 25030:2019, *Software Engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Quality Requirements*, August 2019.
https://www.iso.org/standard/72116.html

[ISO 25060] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) TR 25060:2010, *Systems and software engineering — Systems and software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability: General framework for usability-related information*, July 2010.
https://www.iso.org/standard/35786.html

[ISO 25063] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 25063:2014, *Systems and software engineering – Systems and software product Quality Requirements and Evaluation (SQuaRE) – Common Industry Format (CIF) for usability: Context of use description*, March 2014.
https://www.iso.org/standard/35789.html

[ISO 27001] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27001:2013, *Information technology -- Security techniques -- Information security management systems -- Requirements*, September 2013.
https://www.iso.org/standard/54534.html

[ISO 27002] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27002:2013, *Information technology -- Security techniques -- Code of practice for information security controls*, September 2013.
https://www.iso.org/standard/54533.html

[ISO 27034-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27034-1:2011, *Information technology — Security techniques — Application security — Part 1: Overview and concepts*, November 2011. https://www.iso.org/standard/44378.html

[ISO 27036-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27036-1:2014, *Information technology — Security techniques — Information security for supplier relationships — Part 1: Overview and concepts*, April 2014. https://www.iso.org/standard/59648.html

[ISO 27036-2] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27036-2:2014, *Information technology — Security techniques — Information security for supplier relationships — Part 2: Requirements*, August 2014. https://www.iso.org/standard/82060.html

[ISO 27036-3] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27036-3:2013, *Information technology — Security techniques — Information security for supplier relationships — Part 3: Guidelines for information and communication technology supply chain security*, November 2013. https://www.iso.org/standard/59688.html

[ISO 29110-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) TR 29110-1:2016, *Systems and software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 1: Overview*, June 2016. https://www.iso.org/standard/62711.html

[ISO 29119-1] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 29119-1:2013, *Software Testing: Concepts and Definitions*, September 2013. https://www.iso.org/standard/45142.html

[ISO 29119-2] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 29119-2:2013, *Software Testing: Test Processes*, September 2013. https://www.iso.org/standard/56736.html

[ISO 29119-3] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 29119-3:2013, *Software Testing: Test Documentation*, September 2013. https://www.iso.org/standard/56737.html

[ISO 29119-4] International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 29119-4:2014, *Software Testing: Test Techniques*, December 2015. https://www.iso.org/standard/60245.html

[ISO 29148] International Organization for Standardization /International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (ISO/IEC/IEEE) 29148:2018, *Systems and software engineering — Life cycle processes – Requirements engineering*, November 2018.
https://www.iso.org/standard/72089.html

[ISO 31000] International Organization for Standardization (ISO) 31000:2018, *Risk management – Guidelines*, February 2018.
https://www.iso.org/standard/65694.html

[ISO 42010] International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE), ISO/IEC/IEEE 42010, *Systems and Software Engineering — Architecture description*, December 2011.
https://www.iso.org/standard/50508.html

[ISO 42020] International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE), ISO/IEC/IEEE 42020:2019, *Software, systems and enterprise — Architecture processes*, July 2019.
https://www.iso.org/standard/68982.html

[MILSTD-882E] Department of Defense Standard Practice, *System Safety*, MIL-STD-882E, May 2012.

[SP 800-30] Joint Task Force Transformation Initiative (2012) Guide for Conducting Risk Assessments. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-30, Rev. 1.
https://doi.org/10.6028/NIST.SP.800-30r1

[SP 800-160v2] R. Ross, V. Pillitteri, R. Graubart, D. Bodeau and R. McQuaid (2021) Developing Cyber Resilient Systems: A Systems Security Engineering Approach. (National Institute of Standards and Technology, Gaithersburg, MD), NIST SP 800-160 Volume 2, Revision 1.
https://csrc.nist.gov/publications/detail/sp/800-160/vol-2-rev-1/final

[SP 800-181] R. Petersen, D. Santos, M. C. Smith, K. A. Wetzel G. Witte (2020) Workforce Framework for Cybersecurity (NICE Framework). (National Institute of Standards and Technology, Gaithersburg, MD), NIST SP 800-181 Revision 1.
https://csrc.nist.gov/publications/detail/sp/800-181/rev-1/final

[TCSEC85] Department of Defense (DoD) Standard 5200.28-STD, *Trusted Computer System Evaluation Criteria*, December 1985.
https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/dod85.pdf

OTHER PUBLICATIONS

[Adcock20] Adcock R, Jackson S, Singer J, Hybertson D, "Principles of Systems Thinking," Stevens Institute of Technology, May 2020. https://www.sebokwiki.org/wiki/Principles_of_Systems_Thinking

[Anderson72] Anderson J, *Computer Security Technology Planning Study*, Technical Report ESD-TR-73- 51, Air Force Electronic Systems Division, Hanscom AFB, October 1972. https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/ande72a.pdf

[Anderson20] Anderson R, *Security Engineering: A Guide to Building Dependable Distributed Systems*, 3rd Edition, Wiley, December 2020.

[Ball03] Ball RE, "The Fundamentals of Aircraft Combat Survivability Analysis and Design", 2nd Edition. AIAA Education Series, 2003. https://arc.aiaa.org/doi/book/10.2514/4.862519

[Benjamin14] Benjamin A, et al., "Developing Probabilistic Safety Performance Margins for Unknown and Underappreciated Risks," PSAM-12 International Conf. on Probabilistic Safety and Management, June 2014.

[Bieder20] Bieder C, *The Coupling of Safety and Security - Exploring Interrelations in Theory and Practice*, Springer, 2020. https://link.springer.com/book/10.1007%2F978-3-030-47229-0

[Bryant20] Bryant WD, Ball RE, "Developing the Fundamentals of Aircraft Cyber Combat Survivability: Part 2," *Joint Aircraft Survivability Program Office, Aircraft Survivability Journal*, Spring 2020

[CISA20] Critical Infrastructure Sectors, Department of Homeland Security Cybersecurity and Infrastructure Security Agency. https://www.cisa.gov/critical-infrastructure-sectors

[DOD 2007] Department of Defense, MIL-HDBK-454B, *General Guidelines for Electronic Equipment*, April 2007. https://www.dla.mil/Portals/104/documents/landAndMaritime/v/va/pSMC/documents/lM_MIL_HDBK_454B_151030.pdf

[DOD 2020] Department of Defense, MIL-HDBK-454B, *Mission Engineering Guide*, November 2020. https://ac.cto.mil/wp-content/uploads/2020/12/MEG-v40_20201130_shm.pdf

[DODI 5200] Department of Defense Instruction (DoDI) 5200.39, "Critical Program Information (CPI) Identification and Protection Within Research, Development, Test, and Evaluation (RDT&E)," October 2020. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/520039p.pdf

[DSB 2013] Department of Defense Science Board Task Force Report, *Resilient Military Systems and the Advanced Cyber Threat*, January 2013. https://dsb.cto.mil/reports/2010s/ResilientMilitarySystemsCyberThreat.pdf

[DSB 2017] Department of Defense Science Board, *Task Force on Cyber Deterrence*, February 2017. https://dsb.cto.mil/reports/2010s/DSB-CyberDeterrenceReport_02-28-17_Final.pdf

[FUSE21] Dove R, Willett K, McDermott T, Dunlap H, MacNamara DP, Ocker C. "Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundational Concepts" *INCOSE International Symposium*, July 2021.

[Herley16] Herley C, *Unfalsifiability of Security Claims*, Microsoft Research, Proceedings of the National Academy of Sciences, April 2016.

[INCOSE] International Council On Systems Engineering, *What Is Systems Engineering?* https://www.incose.org/systems-engineering

[INCOSE05] Roedler G, Jones C, *Technical Measurement*, International Council on Systems Engineering, INCOSE TP-2003-020-01, December 2005. https://www.incose.org/docs/default-source/ProductsPublications/technical-measurement-guide---dec-2005.pdf?sfvrsn=4&sfvrsn=4

[INCOSE10] Systems Engineering Measurement Primer, International Council On Systems Engineering INCOSE TP-2010-005-02, November 2010. https://www.incose.org/docs/default-source/ProductsPublications/systems-engineering-measurement-primer---december-2010.pdf

[INCOSE14] System Engineering Handbook—A Guide for System Engineering Life Cycle Processes and Activities, International Council on Systems Engineering, INCOSE TP-2003-002-04, July 2015.

[INCOSE19] Sillitto H, Martin J, McKinney D, Griego R, Dori D, Krob D, Godfrey P, Arnold E, Jackson L, INCOSE-TP-2020-002-06, *Systems Engineering and System Definitions*, July 2019. https://www.incose.org/docs/default-source/default-document-library/incose-se-definitions-tp-2020-002-06.pdf?sfvrsn=b1049bc6_0

[INCOSE20] Guide for Writing Requirements, International Council On Systems Engineering, INCOSE TP-2010-006-03, July 2019. https://connect.incose.org/Pages/Product-Details.aspx?ProductCode=TechGuideWR2019Soft

[Jackson13] Jackson S, Ferris T, “Resilience Principles for Engineered Systems,” *Systems Engineering*, Vol. 16, No. 2, July 2013. https://onlinelibrary.wiley.com/doi/abs/10.1002/sys.21228

[Lampson73] Lampson BW, “A Note on the Confinement Problem,” Communications of the ACM 16, 10, pp. 613-615, October 1973. https://dl.acm.org/doi/10.1145/362375.362389

[Leveson11] Leveson NG, “Engineering a Safer World – Systems Thinking Applied to Safety,” Chapter 14, MIT Press, ISBN 978-0-262-01662-9, 2011. https://direct.mit.edu/books/book/2908/Engineering-a-Safer-WorldSystems-Thinking-Applied

[Maier98] Maier M, “Architecting Principles for Systems-of-Systems,” The Aerospace Corporation, 1998. https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291520-6858%281998%291%3A4%3C267%3A%3AAID-SYS3%3E3.0.CO%3B2-D

[McEvilley15] McEvilley M, “Towards a Notional Framework for Systems Security Engineering,” The MITRE Corporation, NDIA 18th Annual Systems Engineering Conference, October 2015.

[MITRE21] Hild D, McEvilley M, Winstead M, “Principles for Trustworthy Design of Cyber-Physical Systems,” MITRE Technical Report, MTR210263, June 2021.

[Moller08] Moller N, Hansson SO, “Principles of Engineering Safety: Risk and Uncertainty Reduction,” *Reliability Engineering & System Safety*, Vol. 93, No. 6, June 2008.

[NASA07] National Aeronautics and Space Administration (NASA), *Systems Engineering Handbook*, NASA/SP-2007-6105, Revision 1, December 2007. https://www.nasa.gov/sites/default/files/atoms/files/nasa_systems_engineering_handbook.pdf

[NASA11] National Aeronautics and Space Administration (NASA), System Safety Handbook Volume 1: System Safety Framework and Concepts for Implementation, NASA/SP-2010-580, Version 1.0, November 2011. https://ntrs.nasa.gov/api/citations/20120003291/downloads/20120003291.pdf

[NASA14] National Aeronautics and Space Administration (NASA), *System Safety Handbook Volume 2: System Safety Concepts, Guidelines, and Implementation Examples*, NASA/SP-2014-612, Version 1.0, November 2014. https://ntrs.nasa.gov/api/citations/20150015500/downloads/20150015500.pdf

[NASA17] Rinehart DJ, Knight JC, and Rowanhill J, "Understanding What it Means for Assurance Cases to Work," NASA/CR–2017-219582, April 2017. https://catalog.libraries.psu.edu/catalog/20766348

[NASA19] National Aeronautics and Space Administration (NASA), *AdvoCATE: Assurance Case Automation Toolset*, January 2019. https://ti.arc.nasa.gov/tech/rse/research/advocate

[Neumann00] Neumann P, "Practical Architectures for Survivable Systems and Networks," Technical Report, Final Report, Phase Two, Project 1688, SRI International, Menlo Park, California, June 2000. http://www.csl.sri.com/neumann/survivability.html

[Neumann04] Neumann P, "Principled Assuredly Trustworthy Composable Architectures," CDRL A001 Final Report, SRI International, Menlo Park, CA, December 28, 2004. http://www.csl.sri.com/users/neumann/chats4.pdf

[Neumann17] Neumann P, "Fundamental Trustworthiness Principles," 2017.

[NICE Framework] Cybersecurity and Infrastructure Security Agency (CISA), *National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework*. https://niccs.cisa.gov/workforce-development/cyber-security-workforce-framework

[NICE RC] National Initiative for Cybersecurity Education (NICE) Framework Resource Center https://www.nist.gov/itl/applied-cybersecurity/nice/nice-framework-resource-center

[Levin07] Levin T, Irvine C, Benzel T, Bhaskara G, Clark P, and Nguyen T, *Design Principles and Guidelines for Security*, Technical Report NPS-CS-07-014, Naval Postgraduate School, November 2007. https://nps.edu/web/c3o/technical-reports

[Pagani04] Pagani LP, "On the Quantification of Safety Margins," PhD Dissertation, Massachusetts Institute of Technology, September 2004.

[Popek74] Popek G, "The Principle of Kernel Design," in 1974 NCC, AFIPS Cong. Proc., Vol. 43.

[Saleh14] Saleh JH, Marais KB, and Favaro FM, "System safety principles: A multidisciplinary engineering perspective," *Journal of Loss Prevention in the Process Industries*, Vol. 29, 2014.

[Saltzer75] Saltzer JH, Schroeder MD, "The Protection of Information in Computer Systems," in *Proceedings of the IEEE* Vol. 63, No. 9, September 1975. https://www.cs.virginia.edu/~evans/cs551/saltzer

[Saltzer09] Saltzer JH, Kaashoek MF, "Principles of Computer System Design," 2009. https://ocw.mit.edu/resources/res-6-004-principles-of-computer-system-design-an-introduction-spring-2009/online-textbook/readings_open_5_0.pdf

[Saydjari18] Saydjari OS, *Engineering Trustworthy Systems: Get Cybersecurity Design Right the First Time*, McGraw-Hill, August 2018. https://books.apple.com/us/book/engineering-trustworthy-systems-get-cybersecurity-design/id1413527360

[Schroeder72] Schroeder MD, "Cooperation of mutually suspicious subsystems in a computer utility," Ph.D. dissertation, M.I.T., Cambridge, Mass., 1972 https://web.mit.edu/~saltzer/www/publications/TRs+TMs/Multics/TR-104.pdf

[Schroeder77] Schroeder MD, Clark DD, and Saltzer JH, "The Multics Kernel Design Project," in *Proceedings of Sixth ACM Symposium on Operating Systems Principles*, 1977. https://web.mit.edu/Saltzer/www/publications/rfc/csr-rfc-140.pdf

[SEBoK] BKCASE Editorial Board (2019) The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 2.0, ed Cloutier RJ (The Trustees of the Stevens Institute of Technology, Hoboken, NJ). BKCASE is managed and maintained by the Stevens Institute of Technology Systems Engineering Research Center, the International Council on Systems Engineering, and the Institute of Electrical and Electronics Engineers Computer Society. https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)

[Sheard18] Sheard S, Konrad M, Weinstock C, and Nichols W, "A Complexity Measure for System Safety Assurance," in *INCOSE International Symposium*, Adelaide Australia, 2018. https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2334-5837.2017.00373.x

[Simovici08] Simovici DA, Djeraba C, "Partially Ordered Sets," *Mathematical Tools for Data Mining: Set Theory, Partial Orders, Combinatorics*, Springer, 2008.

[Smith12] Smith RE, "A Contemporary Look at Saltzer and Schroeder's 1975 Design Principles," IEEE Security & Privacy, Vol. 10, No. 6, November/December 2012.

[Snyder15] Snyder D, Powers JD, Bodine-Baron E, Fox B, Kendrick L, Powell MH, "Improving the Cybersecurity of U.S. Air Force Military Systems Throughout Their Life Cycles," Rand Corporation, 2015. https://www.rand.org/content/dam/rand/pubs/research_reports/RR1000/RR1007/RAND_RR1007.pdf

[Uchenick05] Uchenick GM, Vanfleet WM, "Multiple Independent Levels of Safety and Security: High Assurance Architecture for MSLS/MLS," *IEEE Military Communications Conference*, 2005, pp. 610-614 Vol. 1.

[Young14] Young W, Leveson NG, "An Integrated Approach to Safety and Security based on Systems Theory," *Communications of the ACM*. Volume 57, Issue 2, 2014, pp. 31-35. https://dl.acm.org/doi/10.1145/2556938

APPENDIX A

GLOSSARY

COMMON TERMS AND DEFINITIONS

Appendix A provides definitions for the engineering and security terminology used within Special Publication 800-160, Volume 1.

abstraction [ISO 24765]	View of an object that focuses on the information relevant to a particular purpose and ignores the remainder of the information.
acquirer [ISO 15288]	Stakeholder that acquires or procures a product or service from a supplier.
acquisition [ISO 15288]	Process of obtaining a system, product, or service.
activity [ISO 15288]	Set of cohesive tasks of a process.
adequate security (systems)	Meets minimum tolerable levels of security, as determined by analysis, experience, or a combination of both; and is as secure as reasonably practicable (i.e., incremental improvement in security would require an intolerable or disproportionate deterioration of meeting other system objectives such as those for system performance, or would violate system constraints).
adverse consequence [ISO 15026-1]	An undesirable consequence associated with a loss.
adversity	The conditions that can cause a loss of assets (e.g., threats, attacks, vulnerabilities, hazards, disruptions, and exposures).
agreement [ISO 15288]	Mutual acknowledgement of terms and conditions under which a working relationship is conducted (e.g., memorandum of agreement or contract).
anomaly [ISO 24765]	Condition that deviates from expectations, based on requirements specifications, design documents, user documents, or standards, or from someone's perceptions or experiences.
anti-tamper [DODI 5200]	Systems engineering activities intended to prevent or delay exploitation of critical program information in U.S. defense systems in domestic and export configurations to impede countermeasure development, unintended technology transfer, or alteration of a system due to reverse engineering. See *tampering*.

architecture [ISO 42010]	Fundamental concepts or properties related to a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution. Refer to *security architecture.*
architecture (system) [ISO 42010]	Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution.
architecture description [ISO 42010]	A work product used to express an architecture.
architecture framework [ISO 42010]	Conventions, principles, and practices for the description of architectures established within a specific domain of application and/or community of stakeholders.
architecture view [ISO 42010]	A work product expressing the architecture of a system from the perspective of specific system concerns.
architecture viewpoint [ISO 42010]	A work product establishing the conventions for the construction, interpretation, and use of architecture views to frame specific system concerns.
artifact [ISO 19014]	Work products that are produced and used during a project to capture and convey information, (e.g., models, source code).
asset [ISO 24765]	Anything that has value to a person or organization. *Note 1:* Assets have interrelated characteristics that include value, criticality, and the degree to which they are relied upon to achieve organizational mission/business objectives. From these characteristics, appropriate protections are to be engineered into solutions employed by the organization. *Note 2:* An asset may be tangible (e.g., physical item such as hardware, software, firmware, computing platform, network device, or other technology components) or intangible (e.g., information, data, trademark, copyright, patent, intellectual property, image, or reputation).
assurance [ISO 15026-1]	Grounds for justified confidence that a claim has been or will be achieved. *Note 1:* Assurance is typically obtained relative to a set of specific claims. The scope and focus of such claims may vary (e.g., security claims, safety claims) and the claims themselves may be interrelated. *Note 2:* Assurance is obtained through techniques and methods that generate credible evidence to substantiate claims.
assurance case [ISO 15026-1]	A reasoned, auditable artifact created that supports the contention that its top-level claim (or set of claims), is satisfied, including systematic argumentation and its underlying evidence and explicit assumptions that support the claim(s).

assurance evidence	The information upon which decisions regarding assurance, trustworthiness, and risk of the solution are substantiated. *Note:* Assurance evidence is specific to an agreed-to set of claims. The security perspective focuses on assurance evidence for security-relevant claims whereas other engineering disciplines may have their own focus (e.g., safety).
availability [ISO 7498-2]	Property of being accessible and usable on demand by an authorized entity.
baseline [IEEE 828]	Formally approved version of a configuration item, regardless of media, formally designated and fixed at a specific time during the configuration item's life cycle. *Note:* The engineering process generates many artifacts that are maintained as a baseline over the course of the engineering effort and after its completion. The configuration control processes of the engineering effort manage baselined artifacts. Examples include stakeholder requirements baseline, system requirements baseline, architecture/design baseline, and configuration baseline.
behavior [ISO 14258 adapted]	The way an entity functions as an action, reaction, or interaction. How a system element, system, or system of systems acts, reacts, and interacts.
body of evidence	The totality of evidence used to substantiate trust, trustworthiness, and risk relative to the system.
claim [ISO 15026-1]	A true-false statement about the limitations on the values of an unambiguously defined property called the claim's property; and limitations on the uncertainty of the property's values falling within these limitations during the claim's duration of applicability under stated conditions.
complex system [INCOSE19]	A system in which there are non-trivial relationships between cause and effect: each effect may be due to multiple causes; each cause may contribute to multiple effects; causes and effects may be related as feedback loops, both positive and negative; and cause-effect chains are cyclic and highly entangled rather than linear and separable.
component	See *system element*.

concept of operations [ANSI G043B]
: Verbal and graphic statement, in broad outline, of an organization's assumptions or intent in regard to an operation or series of operations of new, modified, or existing organizational systems.

Note 1: The concept of operations frequently is embodied in long-range strategic plans and annual operational plans. In the latter case, the concept of operations in the plan covers a series of connected operations to be carried out simultaneously or in succession to achieve an organizational performance objective.

Note 2: The concept of operations provides the basis for bounding the operating space, system capabilities, interfaces, and operating environment.

concept of secure function
: A strategy for achievement of secure system function that embodies proactive and reactive protection capability of the system.

Note 1: This strategy strives to prevent, minimize, or detect the events and conditions that can lead to the loss of an asset and the resultant adverse impact; prevent, minimize, or detect the loss of an asset or adverse asset impact; continuously deliver system capability at some acceptable level despite the impact of threats or uncertainty; and recover from an adverse asset impact to restore full system capability or to recover to some acceptable level of system capability.

Note 2: The concept of secure function is adapted from historical and other secure system concepts such as *Philosophy of Protection*, *Theory of Design and Operation*, and *Theory of Compliance*.

concern [ISO 42020]
: Matter of interest or importance to a stakeholder.

concern (system) [ISO 42010]
: Interest in a system relevant to one or more of its stakeholders.

configuration item [ISO 15288]
: Item or aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process.

consequence [ISO 15026-1]
: Effect (change or non-change), usually associated with an event or condition or with the system and usually allowed, facilitated, caused, prevented, changed, or contributed to by the event, condition, or system.

constraints [ISO 29148]
: Limitation on the system, its design, or its implementation or on the process used to develop or modify a system.

Limitation that restricts the design solution, implementation, or execution of the system.

Note: A constraint is a factor that is imposed on the solution by force or compulsion and may limit or modify the design.

criticality
[CNSSI 4009]
An attribute assigned to an asset that reflects its relative importance or necessity in achieving or contributing to the achievement of stated goals.

customer
[ISO 9000]
Organization or person that receives a product.

cyber-physical system
[ISO 21840 adapted]
A system integrating computation with physical processes whose behavior is defined by both the computational (digital and other forms) and the physical parts of the system.

derived requirement
[ISO 29148]
A requirement deduced or inferred from the collection and organization of requirements into a particular system configuration and solution.

Note 1: The next higher-level requirement is referred to as a "parent" requirement while the derived requirement from this parent is called a "child" requirement.

Note 2: A derived requirement is typically identified during the elicitation of stakeholder requirements, requirements analysis, trade studies or validation.

design
[ISO 24765]
Process to define the architecture, system elements, interfaces, and other characteristics of a system or system element.

[ISO 15288]
Result of the process to be consistent with the selected architecture, system elements, interfaces, and other characteristics of a system or system element.

Note 1: Information, including specification of system elements and their relationships, that is sufficiently complete to support a compliant implementation of the architecture.

Note 2: Design provides the detailed implementation-level physical structure, behavior, temporal relationships, and other attributes of system elements.

design characteristics
[ISO 24765]
Design attributes or distinguishing features that pertain to a measurable description of a product or service.

design margin
[NASA07]
The margin allocated during design based on assessments of uncertainty and unknowns. This margin is often consumed as the design matures.

domain
[ISO 24765 adapted]

A set of elements, data, resources, and functions that share a commonality in combinations of: (1) roles supported, (2) rules governing their use, and (3) protection needs.

Note: Security domains may reflect one or any combination of the following: capability, functional, or service distinctions; data flow and control flow associated with capability, functional, or service distinctions; data and information sensitivity; data and information security; or administrative, management, operational, or jurisdictional authority. Security domains that are defined in the context of one or more of the above items, reflect a protection-focused partitioning of the system that translates to relationships driven by trust concerns.

emergence

The behaviors and outcomes that result from how individual system elements compose to form the system as a whole.

Note: The behavior and outcomes produced by the system are not those of the individual system elements that comprise the system. Rather, the emergent system behavior and outcomes, or properties, result from the composition of multiple system elements.

enabling system
[ISO 15288]

System that supports a system of interest during its life cycle stages but does not necessarily contribute directly to its function during operation.

engineered system
[INCOSE19]

A system designed or adapted to interact with an anticipated operational environment to achieve one or more intended purposes while complying with applicable constraints.

engineering team

The individuals on the systems engineering team with security responsibilities, systems security engineers that are part of the systems engineering team, or a combination thereof.

environment
[ISO 42010]

Context determining the setting and circumstances of all influences upon a system.

event
[ISO 73]

Occurrence or change of a particular set of circumstances.

evidence

Grounds for belief or disbelief; data on which to base proof or to establish truth or falsehood.

Note 1: Evidence can be objective or subjective. Evidence is obtained through measurement, the results of analyses, experience, and the observation of behavior over time.

Note 2: The security perspective places focus on credible evidence used to obtain assurance, substantiate trustworthiness, and assess risk.

facility
[ISO 15288]

Physical means or equipment for facilitating the performance of an action, e.g., buildings, instruments, tools.

incident
[ISO 15288]
Anomalous or unexpected event, set of events, condition, or situation at any time during the life cycle of a project, product, service, or system.

information item
[ISO 24748-6]
Separately identifiable body of information that is produced, stored, and delivered for human use.

information system
[EGOV]
A discrete set of information resources organized for the collection, processing, maintenance, use, sharing, dissemination, or disposition of information.

Refer to *system.*

interface
[ISO 15288]
Wherever two or more logical, physical, or both, system elements or software system elements meet and act on or communicate with each other.

interoperating system
[ISO 15288]
System that exchanges information with the system of interest and uses the information that has been exchanged.

integrity
[ISO 13008]
Quality of being complete and unaltered.

life cycle
[ISO 15288]
Evolution of a system, product, service, project or other human-made entity from conception through retirement.

life cycle model
[ISO 15288]
Framework of processes and activities concerned with the life cycle that may be organized into stages, which also acts as a common reference for communication and understanding.

life cycle security concepts
The processes, methods, and procedures associated with the system throughout its life cycle and provides distinct contexts for the interpretation of system security. Life cycle security concepts apply during program management, development, engineering, acquisition, manufacturing, fabrication, production, operations, sustainment, training, and retirement.

likelihood
[ISO 73]
Chance of something happening.

margin
[MITRE21]
A spare amount or measure or degree allowed or given for contingencies or special situations. The allowances carried to account for uncertainties and risks. See also *design margin* and *operational margin*.

mechanism	A process or system that is used to produce a particular result. The fundamental processes involved in or responsible for an action, reaction, or other natural phenomenon. A natural or established process by which something takes place or is brought about. Refer to *security mechanism*. *Note:* A mechanism can be technology- or nontechnology-based (e.g., apparatus, device, instrument, procedure, process, system, operation, method, technique, means, or medium).
module [ISO 24765]	Program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading. Discrete and identifiable element with a well-defined interface and well-defined purpose or role whose effect is described as relations among inputs, outputs, and retained state.
monitoring [ISO 73]	Continual checking, supervising, critically observing or determining the status in order to identify change from the performance level required or expected.
operational concept [ANSI G043B]	Verbal and graphic statement of an organization's assumptions or intent in regard to an operation or series of operations of a specific system or a related set of specific new, existing, or modified systems. *Note:* The operational concept is designed to give an overall picture of the operations using one or more specific systems, or set of related systems, in the organization's operational environment from the users' and operators' perspectives. See also concept of operations.
operational environment	Context determining the setting and circumstance of all influences upon a delivered system. *Note:* Operational environments include physical (e.g., land, air, maritime, space) and cyberspace contexts.
operational margin [NASA11] [INCOSE19]	The margin that is designed in explicitly to provide space between the worst normal operating condition and the point at which failure occurs (derives from physical design margin).

operator [ISO 15288]	Individual or organization that performs the operations of a system. *Note 1:* The role of operator and the role of user can be vested, simultaneously or sequentially, in the same individual or organization. *Note 2:* An individual operator combined with knowledge, skills, and procedures can be considered as an element of the system. *Note 3:* An operator may perform operations on a system that is operated, or of a system that is operated, depending on whether or not operating instructions are placed within the system boundary.
organization [ISO 9000] [ISO 15288]	Group of people and facilities with an arrangement of responsibilities, authorities and relationships. *Note:* An identified part of an organization (even as small as a single individual) or an identified group of organizations can be regarded as an organization if it has responsibilities, authorities and relationships. A body of persons organized for some specific purpose, such as a club, union, corporation, or society, is an organization.
outcome [ISO 18307]	Result of the performance (or non-performance) of a function or process(es).
party [ISO 15288]	Organization entering into an agreement.
penetration testing [CNSSI 4009]	A test methodology intended to circumvent the security function of a system. *Note:* Penetration testing may leverage system documentation (e.g., system design, source code, manuals) and is conducted within specific constraints. Some penetration test methods use brute force techniques.
problem [ISO 15288]	Difficulty, uncertainty, or otherwise realized and undesirable event, set of events, condition, or situation that requires investigation and corrective action.
process [ISO 9000]	Set of interrelated or interacting activities which transforms inputs into outputs. A program in execution.
process purpose [ISO 15288]	High-level objective of performing the process and the likely outcomes of effective implementation of the process. *Note:* The purpose of implementing the process is to provide benefits to the stakeholders.
process outcome [ISO 12207]	Observable result of the successful achievement of the process purpose.

product
[ISO 9000]

Result of a process.

Note: There are four agreed generic product categories: hardware (e.g., engine mechanical part); software (e.g., computer program); services (e.g., transport); and processed materials (e.g., lubricant). Hardware and processed materials are generally tangible products, while software or services are generally intangible.

project
[ISO 15288]

Endeavor with defined start and finish criteria undertaken to create a product or service in accordance with specified resources and requirements.

Note: A project is sometimes viewed as a unique process comprising co-coordinated and controlled activities and composed of activities from the Technical Management and Technical Processes defined in this document.

protection needs

Informal statement or expression of the stakeholder security requirements focused on protecting information, systems, and services associated with mission/business functions throughout the system life cycle.

Note: Requirements elicitation and security analyses transform the protection needs into a formalized statement of stakeholder security requirements that are managed as part of the validated stakeholder requirements baseline.

qualification
[ISO 12207]

Process of demonstrating whether an entity is capable of fulfilling specified requirements.

quality assurance
[ISO 9000]

Part of quality management focused on providing confidence that quality requirements will be fulfilled.

quality characteristic
[ISO 9000]

Inherent characteristic of a product, process, or system related to a requirement.

Note: Critical quality characteristics commonly include those related to health, safety, security, assurance, reliability, availability, and supportability.

quality management
[ISO 9000]

Coordinated activities to direct and control an organization with regard to quality.

requirement
[ISO 29148]

Statement that translates or expresses a need and its associated constraints and conditions.

[IEEE 610.12, adapted]

A condition or capability that must be met or possessed by a system or system element to satisfy a contract, standard, specification, or other formally imposed documents.

requirements engineering
[ISO 29148]

An interdisciplinary function that mediates between the domains of the acquirer and supplier to establish and maintain the requirements to be met by the system, software or service of interest.

Note: Requirements engineering is concerned with discovering, eliciting, developing, analyzing, verifying, validating, managing, communicating, and documenting requirements.

resource [ISO 15288]
: Asset that is utilized or consumed during the execution of a process.

Note 1: Includes diverse entities such as funding, personnel, facilities, capital equipment, tools and utilities such as power, water, fuel, and communication infrastructures.

Note 2: Resources include those that are reusable, renewable or consumable.

retirement [ISO 15288]
: Withdrawal of active support by the operation and maintenance organization, partial or total replacement by a new system, or installation of an upgraded system.

risk [ISO 73]
: Effect of uncertainty on objectives.

Note 1: An effect is a deviation from the expected, positive or negative. A positive effect is also known as an opportunity.

Note 2: Objectives can have different aspects (such as financial, health and safety, and environmental goals) and can apply at different levels (such as strategic, organization-wide, project, product and process).

Note 3: Risk is often characterized by reference to potential events and consequences, or a combination of these.

Note 4: Risk is often expressed in terms of a combination of the consequences of an event (including changes in circumstances) and the associated likelihood of occurrence.

Note 5: Uncertainty is the state, even partial, of deficiency of information related 1 to understanding or knowledge of an event, its consequence, or likelihood.

risk analysis [ISO 73]
: Process to comprehend the nature of risk and to determine the level of risk.

risk assessment [ISO 73]
: Overall process of risk identification, risk analysis, and risk evaluation.

risk criteria [ISO 73]
: Terms of reference against which the significance of a risk is evaluated.

risk evaluation [ISO 73]
: Process of comparing the results of risk analysis with risk criteria to determine whether the risk and/or its magnitude is acceptable or tolerable.

risk identification [ISO 73]
: Process of finding, recognizing, and describing risks.

risk management [ISO 73]
: Coordinated activities to direct and control an organization with regard to risk.

risk tolerance [ISO 73]
: The organization or stakeholder's readiness to bear the risk after risk treatment in order to achieve its objectives.

Note: Risk tolerance can be influenced by legal or regulatory requirements.

risk treatment [ISO 73]	Process to modify risk.
safety [ISO 12207]	Expectation that a system does not, under defined conditions, lead to a state in which human life, health, property, or the environment is endangered.
security	Freedom from those conditions that can cause loss of assets with unacceptable consequences.
security architecture	A set of physical and logical security-relevant representations (i.e., views) of system architecture that conveys information about how the system is partitioned into security domains and makes use of security-relevant elements to enforce security policies within and between security domains based on how data and information must be protected. *Note:* The security architecture reflects security domains, the placement of security-relevant elements within the security domains, the interconnections and trust relationships between the security-relevant elements, and the behavior and interactions between the security-relevant elements. The security architecture, similar to the system architecture, may be expressed at different levels of abstraction and with different scopes.
security control [OMB A-130]	A mechanism designed to address needs as specified by a set of security requirements.
security domain [CNSSI 4009]	A domain within which behaviors, interactions, and outcomes occur and that is defined by a governing security policy. *Note:* A security domain is defined by rules for users, processes, systems, and services that apply to activity within the domain and activity with similar entities in other domains.
security function	The capability provided by the system or a system element. The capability may be expressed generally as a concept or specified precisely in requirements.
security mechanism [CNSSI 4009]	A method, tool, or procedure that is the realization of security requirements. *Note 1:* A security mechanism exists in machine, technology, human, and physical forms. *Note 2:* A security mechanism reflects security and trust principles. *Note 3:* A security mechanism may enforce security policy and therefore must have capabilities consistent with the intent of the security policy.

security policy
[CNSSI 4009]

A set of rules that governs all aspects of security-relevant system and system element behavior.

Note 1: System elements include technology, machine, and human, elements.

Note 2: Rules can be stated at very high levels (e.g., an organizational policy defines acceptable behavior of employees in performing their mission/business functions) or at very low levels (e.g., an operating system policy that defines acceptable behavior of executing processes and use of resources by those processes).

security relevance

The functions or constraints that are relied upon to, directly or indirectly, to meet protection needs.

Note: the term *security relevance* has been used to differentiate the role of system functions that singularly or in combination, exhibit behavior, produce an outcome, or provide a capability to enforce authorized and intended system behavior or outcomes.

security requirement

A requirement that has security relevance.

security risk
[ISO 73 adapted]

The effect of uncertainty on objectives pertaining to asset loss and the associated consequences.

Note: [ISO 73] defines risk as the effect of uncertainty on objectives. Furthermore, risk can be either positive or negative.

security service
[CNSSI 4009]

A security capability of function provided by an entity.

security specification

The requirements for the security-relevant portion of the system.

Note: The security specification may be provided as a separate document or may be captured with a broader specification.

service
[ISO 15288]

Performance of activities, work, or duties.

Note 1: A service is self-contained, coherent, discrete, and can be composed of other services.

Note 2: A service is generally an intangible product.

specification
[IEEE 610.12]

A document that specifies, in a complete, precise, verifiable manner, the requirements, design, behavior, or other characteristics of a system or component and often the procedures for determining whether these provisions have been satisfied.

Refer to *security specification*.

stage
[ISO 15288]

Period within the life cycle of an entity that relates to the state of its description or realization.

Note 1: As used in this document, stages relate to major progress and achievement milestones of the entity through its life cycle.

Note 2: Stages often overlap.

stakeholder [ISO 15288]	Individual or organization having a right, share, claim, or interest in a system or in its possession of characteristics that meet their needs and expectations.
stakeholder (system) [ISO 42010]	Individual, team, organization, or classes thereof, having an interest in a system.
strength of function	Criterion expressing the minimum efforts assumed necessary to defeat the specified security behavior of an implemented security function by directly attacking its underlying security mechanisms. *Note 1:* Strength of function has as a prerequisite that assumes that the underlying security mechanisms are correctly implemented. The concept of strength of functions may be equally applied to services or other capability-based abstraction provided by security mechanisms. *Note 2:* The term robustness combines the concepts of assurance of correct implementation with strength of function to provide finer granularity in determining the trustworthiness of a system.
susceptibility	The inability to avoid adversity.
supplier [ISO 15288]	Organization or an individual that enters into an agreement with the acquirer for the supply of a product or service. *Note 1:* Other terms commonly used for supplier are contractor, producer, seller, or vendor. *Note 2:* The acquirer and the supplier sometimes are part of the 1 same organization.
system [INCOSE19] [ISO 15288]	An arrangement of parts or elements that together exhibit behavior or meaning that the individual constituents do not. Systems can be *physical* or *conceptual*, or a combination of both. *Note 1:* A system is sometimes considered as a product or as the services it provides. *Note 2:* In practice, the interpretation of its meaning is frequently clarified by the use of an associative noun (e.g., aircraft system). Alternatively, the word “system” is substituted simply by a context-dependent synonym (e.g., aircraft), though this potentially obscures a system principles perspective). *Note 3:* A complete system includes all of the associated equipment, facilities, material, computer programs, services, firmware, technical documentation, and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.
system element [ISO 15288]	Member of a set of elements that constitute a system. *Note:* A system element is a discrete part of a system that can be implemented to fulfill specified requirements.
system of interest [ISO 15288]	System whose life cycle is under consideration.

system of systems [INCOSE14]	System of interest whose system elements are themselves systems; typically, these entail large-scale interdisciplinary problems with multiple, heterogeneous, distributed systems.
[ISO 21839]	Set of systems or system elements that interact to provide a unique capability that none of the constituent systems can accomplish on its own.
system context	The specific system elements, boundaries, interconnections, interactions, and environment of operation that define a system.
system life cycle [IEEE 610.12]	The period of time that begins when a system is conceived and ends when the system is no longer available for use. Refer to *life cycle stages*.
system security requirement	System requirement that has security relevance. System security requirements define the protection capabilities provided by the system, the performance and behavioral characteristics exhibited by the system, and the evidence used to determine that the system security requirements have been satisfied. *Note 1:* Due to the complexity of system security, there are several types and purposes of system security requirements. These include: (1) structural security requirements that express the passive aspects of the protection capability provided by the system architecture, and (2) functional security requirements that express the active aspects of the protection capability provided by the engineered features and devices (e.g., security mechanisms, inhibits, controls, safeguards, overrides, and countermeasures). *Note 2:* Each system security requirement is expressed in a manner that makes verification possible via analysis, observation, test, inspection, measurement, or other defined and achievable means.
systems engineering [INCOSE19]	A transdisciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts, and scientific, technological, and management methods.
[ISO 24765]	Interdisciplinary approach governing the total technical and managerial effort required to transform a set of stakeholder needs, expectations, and constraints into a solution and to support that solution throughout its life.
systems security engineer	Individual that practices the discipline of systems security engineering, regardless of their formal title. Additionally, the term *systems security engineer* refers to multiple individuals operating on the same team or cooperating teams.

systems security engineering

A transdisciplinary and integrative approach to enable the successful secure realization, use, and retirement of engineered systems, using systems, security, and other principles and concepts, as well as scientific, technological, and management methods. Systems security engineering is a subdiscipline of systems engineering.

tampering [CNSSI 4009]

An intentional but unauthorized act resulting in the modification of a system, components of systems, its intended behavior, or data.

task [ISO 15288]

Required, recommended, or permissible action, intended to contribute to the achievement of one or more outcomes of a process.

threat [CNSSI 4009]

An event or condition that has the potential for causing asset loss and the undesirable consequences or impact from such loss.

Note: The specific causes of asset loss, and for which the consequences of asset loss are assessed, can arise from a variety of conditions and events related to adversity, typically referred to as disruptions, hazards, or threats. Regardless of the specific term used, the basis of asset loss constitutes all forms of intentional, unintentional, accidental, incidental, misuse, abuse, error, weakness, defect, fault, and/or failure events and associated conditions.

traceability [ISO 29110-1]

Discernible association among two or more logical entities, such as requirements, system elements, verifications, or tasks.

traceability analysis

The analysis of the relationships between two or more products of the development process conducted to determine that objectives have been met or that the effort represented by the products is completed.

Note: A requirements traceability analysis demonstrates that all system security requirements have been traced to and are justified by at least one stakeholder security requirement, and that each stakeholder security requirement is satisfied by at least one system security requirement.

traceability matrix [IEEE 610.12]	A matrix that records the relationship between two or more products of the development process (e.g., a matrix that records the relationship between the requirements and the design of a given software component). *Note 1:* A traceability matrix can record the relationship between a set of requirements and one or more products of the development process and can be used to demonstrate completeness and coverage of an activity or analysis based upon the requirements contained in the matrix. *Note 2:* A traceability matrix may be conveyed as a set of matrices representing requirements at different levels of decomposition. Such a traceability matrix enables the tracing of requirements stated in their most abstract form (e.g., statement of stakeholder requirements) through decomposition steps that result in the implementation that satisfies the requirements.
trade-off [ISO 15288]	Decision-making actions that select from various requirements and alternative solutions on the basis of net benefit to the stakeholders.
trade-off analysis	Determining the effect of decreasing one or more key factors and simultaneously increasing one or more other key factors in a decision, design, or project.
trust [MITRE21]	A belief that an entity meets certain expectations and therefore can be relied upon. *Note:* The term belief implies that trust may be granted to an entity whether the entity is trustworthy or not.
trust relationship	An agreed upon relationship between two or more system elements that is governed by criteria for secure interaction, behavior, and outcomes relative to the protection of assets. *Note:* This refers to trust relationships between system elements implemented by hardware, firmware, and software.
trustworthiness [Neumann04]	Worthy of being trusted to fulfill whatever critical requirements may be needed for a particular component, subsystem, system, network, application, mission, enterprise, or other entity. *Note:* From a security perspective, a trustworthy system is a system that meets specific security requirements in addition to meeting other critical requirements.
trustworthy	The degree to which the security behavior of a component is demonstrably compliant with its stated requirements.
user [ISO 25010]	Individual or group that interacts with a system or benefits from a system during its utilization. *Note:* The role of user and the role of operator are sometimes vested, simultaneously or sequentially, in the same individual or organization.

validation [ISO 9000]	Confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled. *Note:* A system is able to accomplish its intended use, goals and objectives (i.e., meet stakeholder requirements) in the intended operational environment. The right system was built.
verification [ISO 9000]	Confirmation, through the provision of objective evidence, that specified requirements have been fulfilled. *Note:* Verification is a set of activities that compares a system or system element against the required characteristics. This includes, but is not limited to, specified requirements, design description, and the system itself. The system was built right.
verification and validation [IEEE 610.12]	The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements.
view [ISO 24774]	Representation of a whole system from the perspective of a related set of concerns. *Note:* A view can cover the entire system being examined or only a part of that system.
viewpoint [ISO 24774]	Specification of the conventions for constructing and using a view.
vulnerability	A weakness that can be exploited or triggered to produce an adverse effect. The inability to withstand adversity. *Note:* Vulnerability can exist in anywhere throughout the life cycle of a system, such as in the CONOPS, procedures, processes, requirements, design, implementation, utilization, and sustainment of the system.

APPENDIX B

ACRONYMS

COMMON ABBREVIATIONS

CNSS	Committee on National Security Systems
DoD	Department of Defense
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
ISO	International Organization for Standardization
IT	Information Technology
NIST	National Institute of Standards and Technology
NDI	Non-Developmental Item
SecDOP	Security Design Order of Precedence
SSE	Systems Security Engineering

APPENDIX C

SECURITY POLICY AND REQUIREMENTS

CRITICAL ELEMENTS FOR BUILDING TRUSTWORTHY SECURE SYSTEMS

This appendix addresses security requirements and policy considerations in support of Chapter Three, Appendix D, and Appendix E but is not a complete tutorial on either. This appendix also discusses the rules and scope of control for security policy (Section C.1); stakeholder and system security requirements (Section C.2); secure and non-secure system states and modes (Section C.3); and the relationship among security requirements, policy, and mechanisms (Section C.4).

C.1 SECURITY POLICY

A security policy is a set of rules (Section C.1.1) that governs behavior within a defined scope of control (Section C.1.2). The term *security policy* is used in different ways including: (1) *security policy objectives*, (2) *organizational security policy*, and (3) *system security policy*. Security policies have a variety of contexts, authorities, scopes, and purposes as described in Section C.1.2, and typically form hierarchical relationships (e.g., security policy objectives subsume organizational security policy, which in turn subsumes system security policy).[65]

C.1.1 Rules

Security policy rules are stated in terms of subjects (i.e., active entities), objects (i.e., passive entities), and the operations that subjects can perform or invoke on objects.[66] The rules for each security policy govern *subject-to-object* behaviors and outcomes. The rules must be accurate, consistent, compatible, and complete with respect to stakeholder objectives for the defined scope of control. Otherwise, gaps in the desired governed behavior will occur.

C.1.2 Scope of Control

Security policies reflect and are derived from laws, directives, regulations, life cycle concepts,[67] requirements, or stakeholder objectives. Each security policy includes a *scope of control* that establishes the bounds within which the policy applies. Security policy objectives, organizational security policy, and system security policy typically have a specific scope of applicability as follows:

- **Security Policy (Protection) Objectives:** Policy objectives capture what is to be achieved or a preferred state. Security policy objectives include assets[68] to be protected, a statement of intent to protect the assets within the specific scope of stakeholder responsibility, and the scope of protections. Security policy objectives are the basis for the derivation of all other security policy forms.

[65] Note that *policy*, at the organization and system level, may be plural in practice and captured across multiple entities for management purposes.

[66] Active entities exhibit behavior (e.g., a process in execution) while passive entities do not (e.g., data, file).

[67] Life cycle concepts include operation, sustainment, evolution, maintenance, training, startup, and shutdown.

[68] Implicitly or explicitly.

- **Organizational Security Policy:** Organizational policy is the set of rules[69] that regulate how an organization achieves its objectives. To be meaningful, the rules provide individuals with a reasonable ability to determine whether their actions either violate or comply with the policy. Organizational security policy defines the behavior of individuals in performing their missions and business functions and is used for the development of processes and procedures.
- **System Security Policy:** System security policy specifies what the security capability of the system is expected to do. It is the set of restrictions and properties that specifies how a system enforces or contributes to the enforcement of an organizational security policy.

Security policy goes through an iterative refinement process that decomposes an abstract statement of security policy into more specific statements of security policy. This occurs in parallel with security requirements allocation and the decomposition of requirements as the system design matures. Figure C-1 illustrates security policy allocation across the organization.

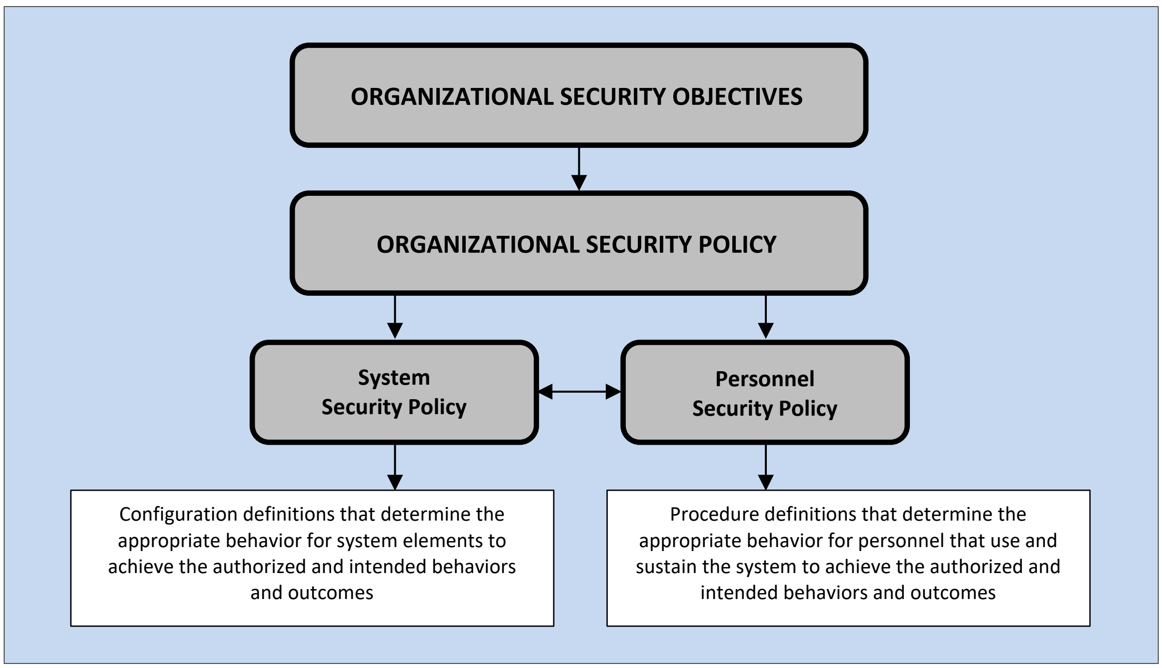


FIGURE C-1: ALLOCATION OF SECURITY POLICY RESPONSIBILITIES

C.2 REQUIREMENTS

A *requirement* is a statement that translates or expresses a specific need and its associated constraints and conditions [ISO 29148].[70] *Security requirements* translate or express protection needs (Section 2.3.7), associated constraints, and associated conditions. The constraints also reflect concerns about the system functions, system architecture, and design to ensure that they are specified in a manner that avoids and reduces susceptibilities, defects, flaws, and weaknesses (Section 2.3.8) and is consistent with the needs of active security functions.

[69] The rules may be captured in laws and practices.

[70] General requirements and definition processes are described in sources such as [ISO 29148] and [INCOSE20].

Requirements can be categorized as: (1) *stakeholder requirements* that address the need to be satisfied in a design-independent manner; and (2) *system requirements* that express the specific solution that will be delivered (design-dependent manner). Figure C-2 illustrates the two types of requirements and their relationship to the verification and validation of the system.

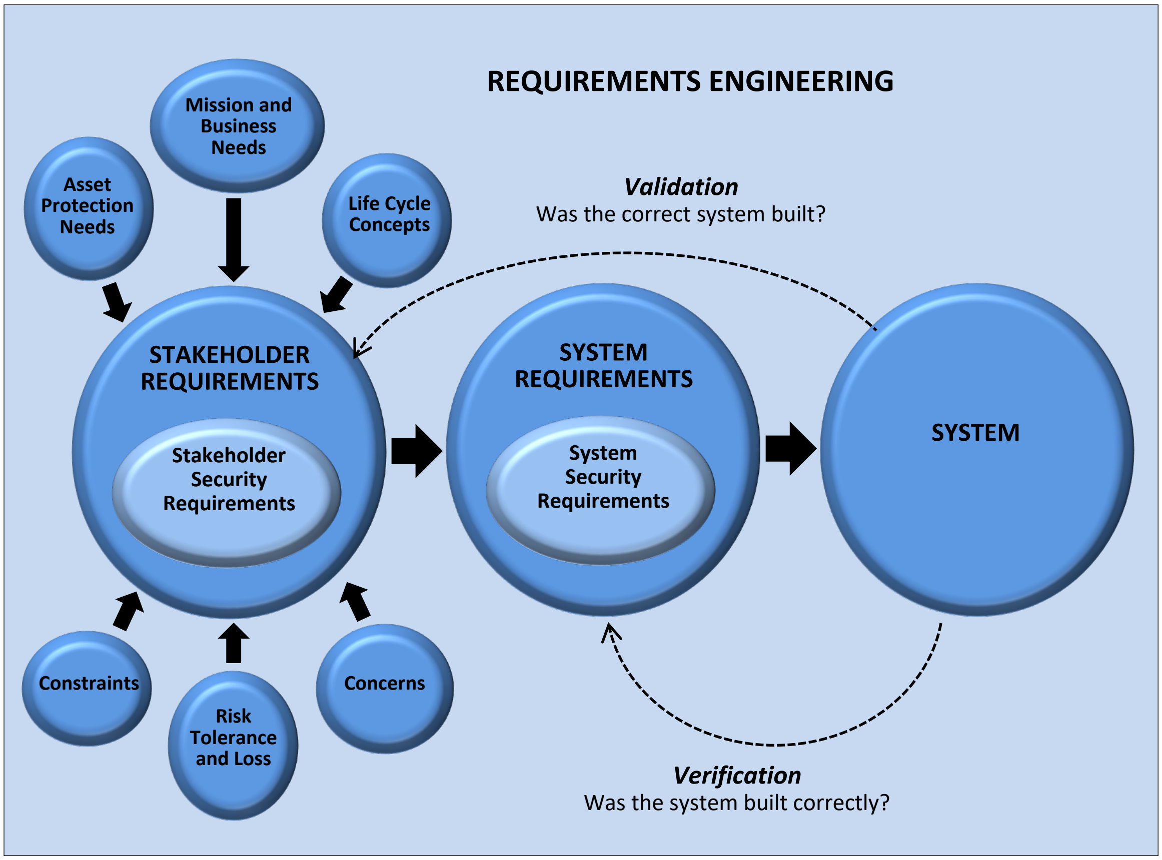


FIGURE C-2: STAKEHOLDER AND SYSTEM REQUIREMENTS

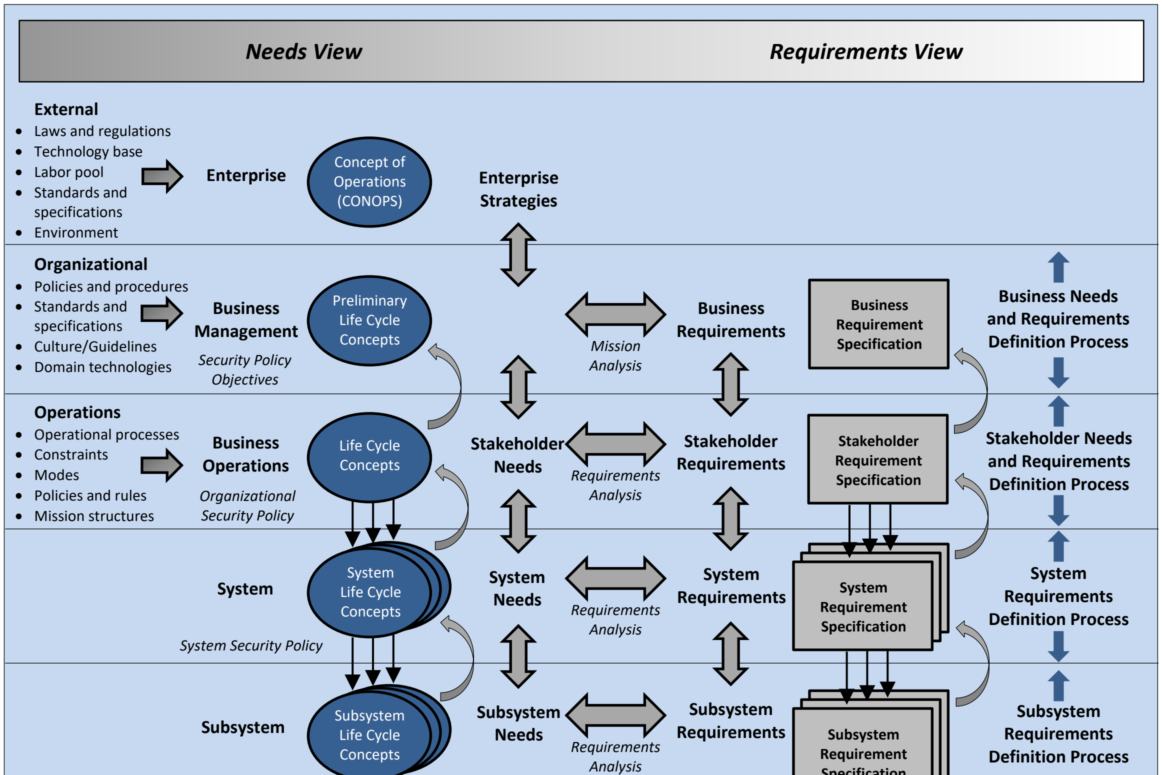
Security requirements and security-relevant constraints and conditions on other requirements are informed by various items, such as those pictured in Figure C-3.

C.2.1 Stakeholder Security Requirements

Stakeholder security requirements are those stakeholder requirements that are security relevant. Stakeholder security requirements specify:

- The protection needed for the mission or business, data, information, processes, functions, humans, and system assets
- The roles, responsibilities, and security-relevant actions of individuals who perform and support the mission or business processes
- The interactions between the security-relevant solution elements
- The assurance that is to be obtained in the security solution

Systems security considerations within activities and tasks (such as those described in Chapter Three) provide the security perspective to ensure that the appropriate stakeholder security requirements are included in the stakeholder requirements and that the stakeholder security requirements are consistent with all other stakeholder requirements.



Adapted from [INCOSE20].

FIGURE C-3: ENTITIES THAT AFFECT SECURITY REQUIREMENT DEVELOPMENT

C.2.2 System Security Requirements

System requirements specify the technical view of a system or solution that meets the specified stakeholder needs. The system requirements are a transformation of the validated stakeholder requirements. System requirements specify what the system or solution must do to satisfy the stakeholder requirements. *System security requirements* are those system requirements that are security relevant. These requirements define:

- The protection capabilities provided by the security solution
- The performance and behavioral characteristics exhibited by the security solution
- Assurance processes, procedures, and techniques
- Constraints on the system and the processes, methods, and tools used to realize the system
- The evidence required to determine the system security requirements have been satisfied[71]

[71] Each system security requirement is expressed in a manner that makes verification possible via observation, analysis, test, inspection, measurement, or other defined and achievable means.

Due to the complexity of system security, there are several types and purposes of system security requirements. These include: (1) *structural security requirements* that express the passive aspects of the protection capability provided by the system architecture, and (2) *functional security requirements* that express the active aspects of the protection capability provided by engineered features and devices (e.g., security mechanisms, controls, safeguards, inhibits, overrides, and countermeasures). The decomposition of system security requirements is accomplished as part of the system requirements decomposition and is to be consistent with the different levels of hierarchical abstraction and forms of the system requirements.

C.3 SYSTEM STATES—SECURE AND NON-SECURE

Systems once implemented will have states which may be secure or nonsecure. Policy and requirements reflect these states. In Section 2.3.4, the definition of security was interpreted to capture what is meant by a secure system:

> *A secure system is a system that – for all of its identified states, modes, and transitions – ensures that only the authorized intended behaviors and outcomes occur, thereby providing freedom from those conditions, both intentionally/with malice and unintentionally/without malice, that can cause a loss of assets with unacceptable consequences.*

This interpretation expresses an **ideal** that captures the essential aspects of what it means to achieve system security. These aspects include:

- Enabling the delivery of the required capability despite intentional and unintentional forms of adversity.
- Enforcing constraints to ensure that only the desired behaviors and outcomes associated with the required capability are realized while satisfying the first aspect.
- Enforcing constraints based on a set of rules to ensure that only authorized human-to-machine and machine-to-machine interactions and operations are allowed to occur while satisfying the second aspect.

The system security policy and system requirements reflect that the set of all possible system states may be partitioned into the set of secure states (i.e., what states are allowed) and the set of nonsecure states (i.e., what states are not allowed). A secure system is, therefore, a system that begins execution in a secure state and cannot transition to a nonsecure state. That is, every state transition results in the same secure state or another secure state. Each state transition must also be secure. Figure C-4 illustrates these "idealized" secure system state transitions.

While it is theoretically possible to engineer such an idealized system, it is impractical to do so. Therefore, security policies and requirements should include additional states and supporting state transitions that reflect the key principles of *Protective Failure* and *Protective Recovery*. Protective failure requires the ability to: (1) detect that the system is in a nonsecure state, and (2) detect a transition that will place the system into a nonsecure state to avoid the propagation of new failure.

Protective failure calls for responsive and corrective actions. It includes transitioning to a secure halt state with a protected recovery to allow for continuation of operations in a reconstituted, reconfigured, or alternative secure operational mode. Other stakeholder objectives may also necessitate the continuation of operations in a less-than-fully-secure state. The policy and requirements should reflect such necessities. Protective recovery requires the ability to effect reactive, responsive, or corrective action to securely transition from a nonsecure state to a secure state (or a less insecure state). The secure state achieved after completion of protective recovery actions includes those actions that limit or prevent any further state transition and those that constitute some type of degraded mode, operation, or capability.

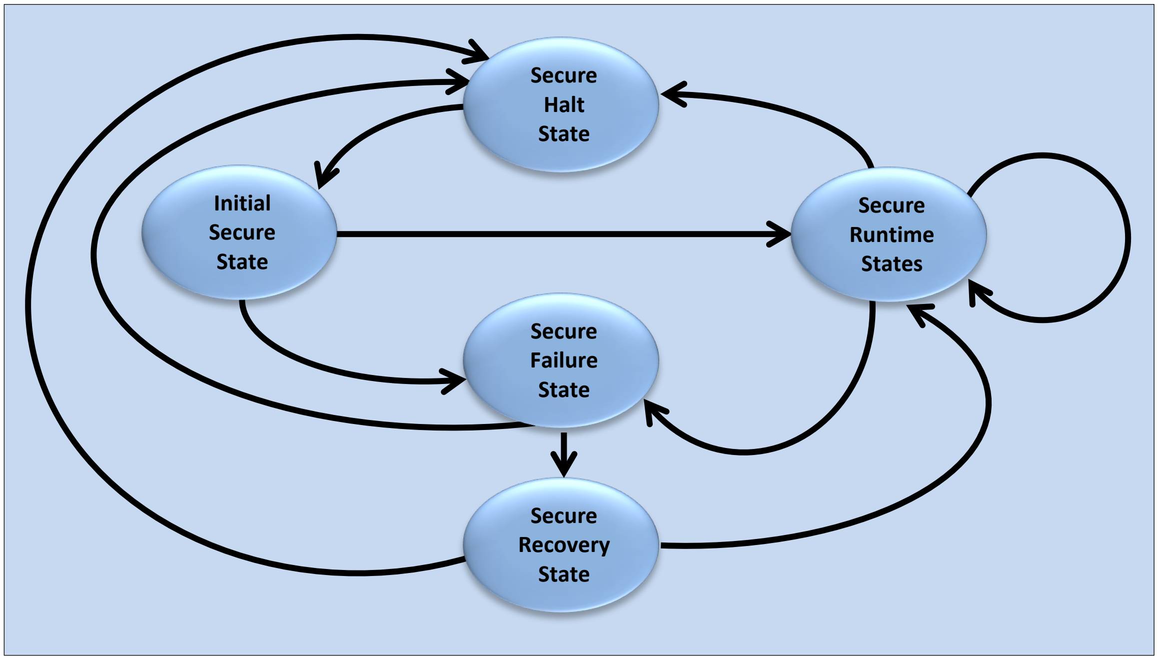


FIGURE C-4: IDEALIZED SECURE SYSTEM STATE TRANSITIONS

C.4 DISTINGUISHING REQUIREMENTS, POLICY, AND MECHANISMS

The terms *requirements*, *policy*, and *mechanisms* are often used in abstract manners that allow them to be considered as synonyms. However, when used in the context of the engineering of trustworthy secure systems, these terms are distinct in their meaning and importance to specifying, realizing, utilizing, and sustaining systems in a trustworthy secure manner.

The security policy states the behavior that is necessary to achieve a secure condition, whereas a security mechanism is a means to achieve the necessary behavior. The distinction between security policy and security mechanism extends to differentiating security requirements from security policy. Security requirements specify the capability, behavior, and quality attributes exhibited and possessed by security mechanisms as well as constraints on each. Security policy specifies how the security mechanisms must behave in some operational context and the constraints on those behaviors. From the system standpoint, a human is a system element and may serve as a security mechanism. Therefore, the human is expected to behave as stated by relevant security policy and security requirements.

Requirements, policies, and mechanisms have an important dependency relationship. System security requirements specify the capabilities and behaviors that a security mechanism is able to provide. A security policy specifies the particular aspects that a mechanism must enforce to achieve organizational objectives. This means that a secure system cannot be achieved if the security requirements do not fully specify the minimal capability necessary to enforce the security policy. It also means that the satisfaction of requirements alone does not result in a secure system. Verification and validation activities must be accomplished separately and coordinated to ensure the individual and combined correctness and effectiveness of the requirements and policy.

Figure C-5 illustrates the significance of the consistency relationship that must be maintained across interacting security requirements, security policy, and security mechanisms.

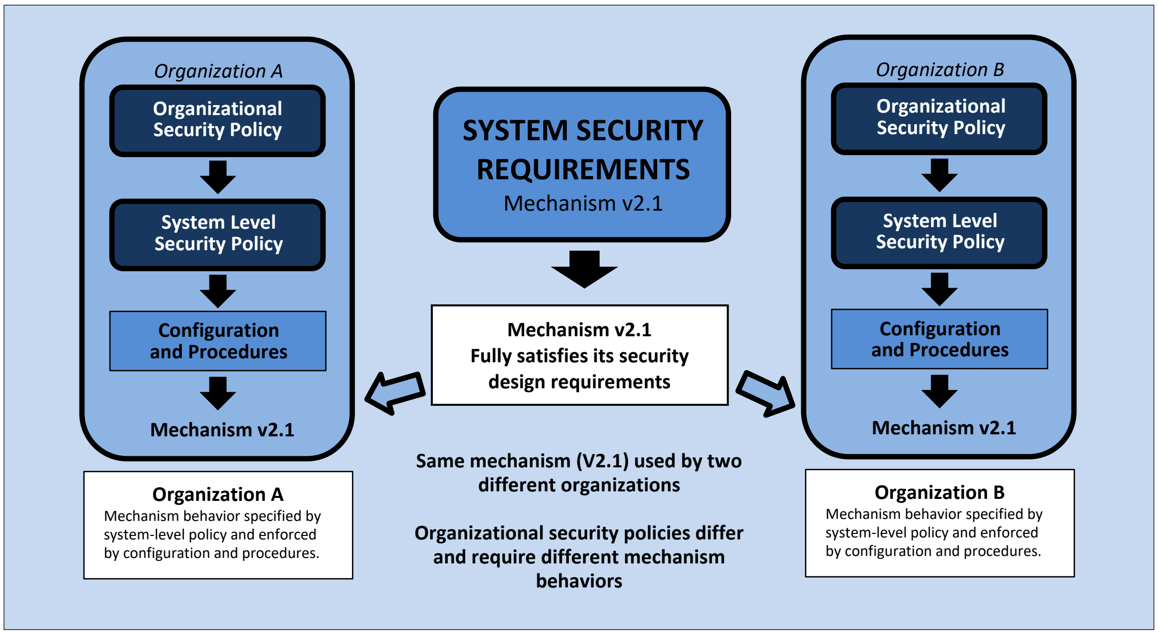


FIGURE C-5: RELATIONSHIP BETWEEN MECHANISMS AND SECURITY POLICY ENFORCEMENT

Any security mechanism that fully satisfies its system security requirements may be deemed capable of enforcing the security policy that is defined for two different organizations. Each organization will use the same mechanism and configure it to behave in a manner that enforces the rules of their organizational security policy. However, if the organizations were to switch mechanisms and keep the same configuration of the mechanism, they would achieve uncertain results (unless their security policy objectives required the exact same configuration of the mechanism). From this, the following conclusions may be drawn:

- Requirements determine the capability for security mechanisms
- Security policy determines the behavior that is deemed “secure” behavior
- For a mechanism to be deemed secure, the requirements for the capability of the mechanism must be consistent with the security policy enforcement rules; the mechanism must satisfy the security requirements; and the mechanism must be configured to behave in a manner defined by the organizational security policy.

APPENDIX D

TRUSTWORTHY SECURE DESIGN

FOUNDATIONAL CONCEPTS FOR THE TRUSTWORTHY SECURE DESIGN OF SYSTEMS

This appendix discusses the approach and considerations for application of the elements of a trustworthy secure system design. This includes a discussion of the system's authorized and intended behaviors and outcomes, the security design order of precedence, and the functional design and trade space considerations.

A principled and effective system design is necessary for trustworthiness. The principled basis and the effectiveness of the design is supported by evidence, thereby making the resultant system trustworthy. The trustworthy secure design concepts described in this appendix provide a balanced and integrated approach that optimally protects against asset loss.

The content in this appendix is supplemented by an in-depth discussion of the principles for trustworthy secure design in Appendix E and the concepts of trustworthiness and assurance in Appendix F. The application of the principles should be planned for, appropriately scoped, and revisited throughout the system life cycle and engineering effort. The principles provide a sound basis for reasoning about a system and permit a demonstration of system trustworthiness through *assurance* based on relevant and credible evidence.

D.1 DESIGN APPROACH FOR TRUSTWORTHY SYSTEMS

The design approach for engineering trustworthy secure systems is intended to establish and maintain the ability to deliver system capabilities at an acceptable level of performance[72] while minimizing the occurrence and extent of loss. The approach provides a system structure for optimal employment of the tactical engineered features and devices.[73] [74] The system design must provide the intended behaviors and outcomes, avoid the unintended behaviors and outcomes, prevent loss, and limit loss when it occurs. A trustworthy secure design includes a margin[75] and a situational awareness capability[76] to account for the unknowns and uncertainty inherent in the system and its operational environment, as well as related adversity.

---

[72] An acceptable level of performance lies between the minimum threshold of acceptability and the objective of maximum performance. The level of acceptable performance may vary across operational or system states and modes (e.g., patrolling in clear weather versus severe weather conditions), may vary across contingency conditions (e.g., normal, degraded), and may be subject to operational priorities (e.g., search and rescue, manhunt).

[73] The term *tactics* refers to a specific means to accomplish an action. Tactics focus on *how* to accomplish the action (e.g., using engineered features and devices, including security controls, to react to a threat). This is in contrast to the term *strategy*, which takes a broader view and focuses on *what* to accomplish (e.g., a design approach for trustworthy secure systems) [Young14].

[74] [Snyder15] postulates that "poor systems security engineering is very difficult to mitigate by overlaying security controls, whereas security controls overlaid on a sound, secure design can be quite effective."

[75] The term *margin* refers to a spare amount, measure, or degree allowed or given for contingencies or special situations. The allowances are carried to account for uncertainties and risks. In general, there are two types of margins used in systems engineering: *design margin* and *operational margin*. See the design principle of *Loss Margins*.

[76] A *situational awareness* capability includes detecting pending and actual failure (e.g., by crossing the threshold of the margins that have been established). See the design principle of *Anomaly Detection*.

The design approach includes the following elements:[77]

- Define the intended behaviors and outcomes for the system.[78]
- Identify the system states and conditions that reflect the intended behaviors and outcomes.
- Identify the system states and conditions that potentially lead to loss in the system.
- Engineer to prevent loss to the extent practicable (preferred), and limit the loss that does occur (where, when, and to the extent necessary and practicable).

Iterate the above elements to address how the functions that serve to prevent or limit loss may fail due to intentional or unintentional reasons.

Figure D-1 illustrates the steps in the design approach in the context of the *Systems Security Engineering Framework* described in Section 2.5.

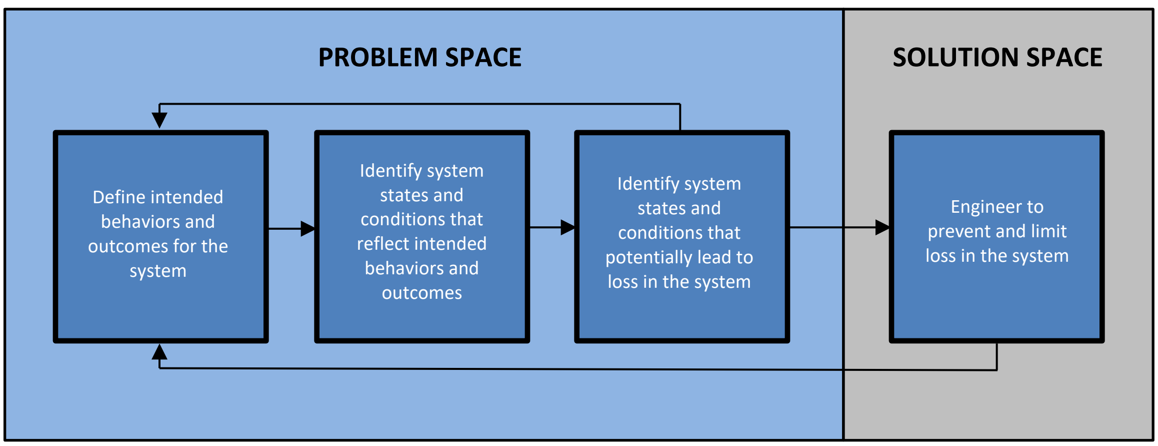


FIGURE D-1: DESIGN APPROACH IN A SYSTEMS SECURITY ENGINEERING FRAMEWORK

D.2 DESIGN FOR BEHAVIORS AND OUTCOMES

A system is to deliver the required capability at a specified level of performance. The system capability is reflected in its behaviors and outcomes. The design goal is to provide capabilities that are authorized and intended. However, the system can also deliver a capability that is not authorized or intended. This possibility exists due to the concept of *emergence*. Emergence refers to the behaviors and outcomes that result from how individual system elements compose to form the system as a whole. That is, the behavior and outcomes produced by the system are not those of the individual system elements that comprise the system. Rather, the emergent system behavior and outcomes, or properties, result from the composition of multiple system elements (see trustworthy secure design principle *Structured Decomposition and Composition* and Figure 4).

[77] These steps are useful in applying a *system control* concept for any loss-relevant emergent property (e.g., safety, security, resilience).

[78] This flow iterates through systems engineering as the system is decomposed. Subsequent iterations of this same approach would apply within the elements that comprise the system of interest (i.e., the subsystems, assemblies, and components).

Additionally, while the emergent system properties sought are desired and productive, there are emergent properties that are not desired or productive. Such properties can produce unknown, unforeseen, or adverse effects. The engineering of trustworthy secure systems seeks to deliver only the desired and productive emergent properties of the system because trustworthiness judgments are based on the expectation that the system can satisfy the stated capability needs. To achieve this, the design must address emergence at all levels of system abstraction in terms of how the system is decomposed into its constituent elements and how those elements compose to produce the system (see the design principle of *Compositional Trustworthiness*).

SECURITY AS AN EMERGENT SYSTEM PROPERTY

The objective of security as an emergent system property is to achieve *only* the authorized and intended system behaviors and outcomes. This requires a fundamental understanding of how individual system elements are composed into the system as a whole. Systems are designed from that basis of understanding to limit the emergent behaviors and outcomes that are not specified (including desired unspecified and undesired unspecified behaviors and outcomes).

Both *proactive* and *reactive* aspects are considered as part of an integrated and balanced engineering approach to defining the authorized and intended behaviors and outcomes needed to address protection needs. The proactive aspect of the engineering effort addresses actions taken to prevent and limit loss before the event occurs, while the reactive aspect addresses actions taken to limit loss and its effects once an event has occurred. The proactive aspect recognizes the conditions where loss may occur and addresses the scenarios before loss occurs. If the loss does occur, the results are limited due to actions taken in advance. It is independent of any specific knowledge of attacks and attacker objectives and is focused on what is possible in the system's life cycle.

The reactive aspect of the engineering effort recognizes that new, unanticipated, and otherwise unforeseen adverse consequences will occur despite the proactive planning and institution of means and methods to control loss and the extent of its consequences. The reactive aspect enables informed operational decision-making once the system is in use and a loss condition occurs, proactively giving operations the ability to deal with the loss condition and to better deal with the loss. The reactive aspect complements the proactive aspect by providing an informed basis and means for an external entity (e.g., a human operator or system of systems) to act when failures occur. In essence, the reactive aspect is a proactive engineering activity about providing a *reactive capability*.

The proactive and reactive aspects must be balanced across all assets, stakeholders, concerns, and objectives. Achieving such balance requires that security objectives be established and that requirements elicitation and analysis be conducted to unambiguously and clearly ascertain the scope of security in terms of addressing failure and the associated consequences in its proactive and reactive aspects. Figure D-2 illustrates the balanced design strategy for achieving trustworthy secure systems.

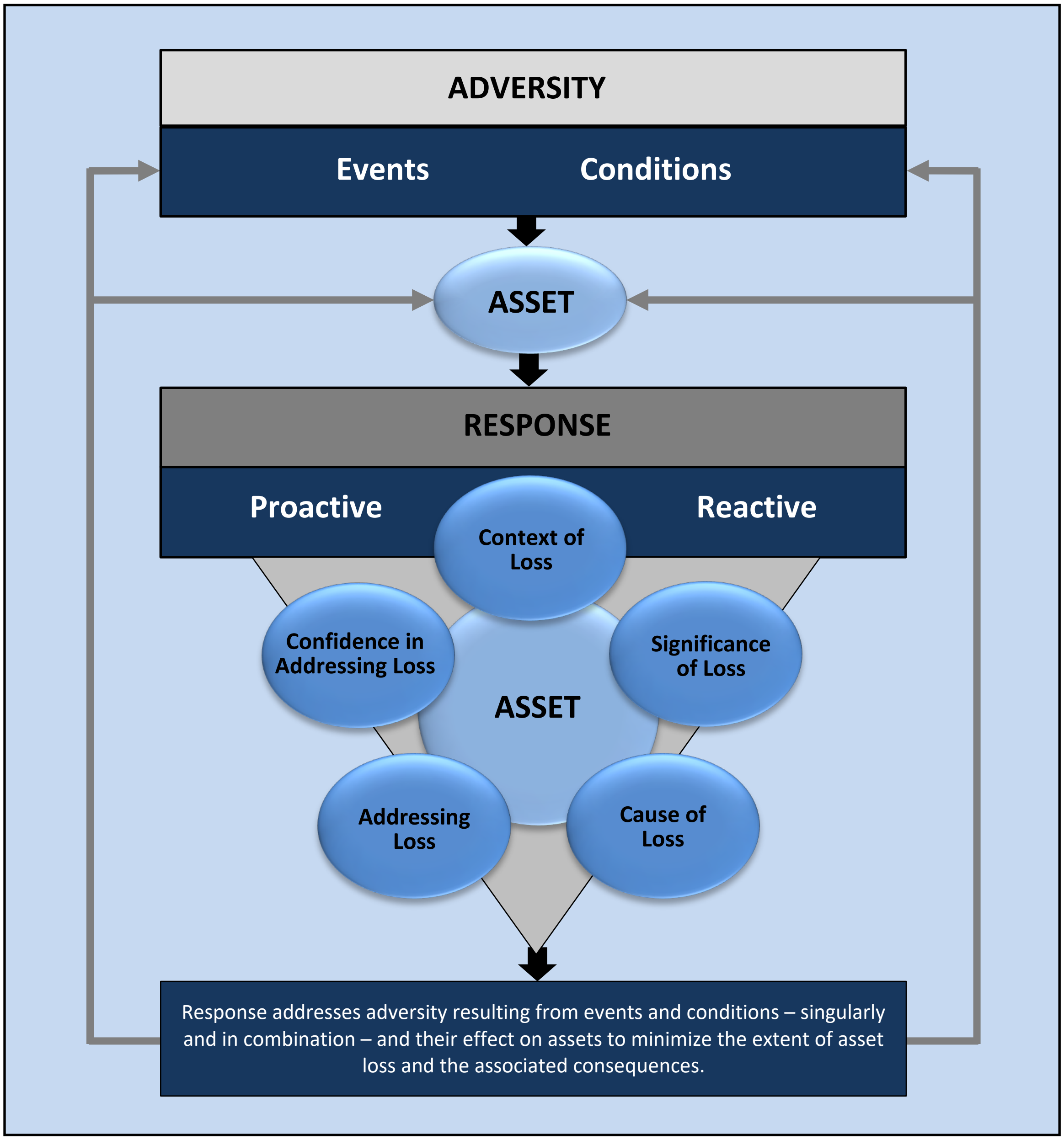


FIGURE D-2: BALANCED DESIGN STRATEGY FOR ACHIEVING TRUSTWORTHY SECURE SYSTEMS

D.3 SECURITY DESIGN ORDER OF PRECEDENCE

The security design order of precedence (SecDOP)[79] is part of a design approach that uses passive architectural features to provide the structure for the employment of engineered features and devices. SecDOP reflects a design goal to eliminate the design basis for loss potential. Using a principled and assured engineering approach, the SecDOP eliminates susceptibility, hazard, and vulnerability to the extent practicable, thereby eliminating the

[79] The *security design order of precedence* is inspired by the *System Safety Design Order of Precedence*, an optimized design approach for system safety described in [MILSTD-882E].

associated risk. For those cases in which susceptibility, hazard, or vulnerability cannot be eliminated, the SecDOP reduces the loss potential (e.g., occurrence, impact) to the lowest acceptable level within the constraints of cost, schedule, and performance. The SecDOP identifies the design options and lists those options in order of decreasing effectiveness, thus enabling a maximized return on investment.

The SecDOP acts as follows:

1. **Eliminate the potential for loss through design selection.**

   Susceptibility, hazard, and vulnerability are eliminated by selecting a design or material alternative that completely removes the susceptibility, hazard, and vulnerability and thus prevents loss.

   Example: The design selected for a *system function of interest* minimizes the number of interfaces to other systems (i.e., external interfaces) and the number of internal interfaces (i.e., interfaces with no connection to other systems). The minimization of interfaces (both external and internal) is determined in consideration of the interface needs of *all system functions* and results in an across-the-board optimization that does not overly constrain the design for the *system function of interest.* That is, the design results in less susceptibility, hazard, and vulnerability than a design that incorporates additional and unnecessary internal and external interfaces.

   Note: The design selection to control loss is accomplished to accommodate the need for mechanisms that provide mediated access and trusted communication as these engineered features and devices are necessary for a secure system.

2. **Reduce the potential for loss through design alteration.**

   If adopting an alternative design or material to eliminate susceptibility, hazard, and vulnerability is not feasible, consider design changes or material selection that would reduce the frequency, potential, severity, and/or extent of loss caused by the susceptibility, hazard, or vulnerability.

   Example: The selected design for the *system function of interest* has susceptibility, hazard, and vulnerability due to the system-level design trades made to satisfy the requirements for *all system functions*, emergence, and the limits of certainty. In response to these conditions, the design might consider functional domains, defense-in-depth layering, redundancy, and other approaches to further reduce susceptibility, hazard, and vulnerability.

   Note: The design alteration to control loss is accomplished to accommodate the need for mechanisms that provide mediated access and trusted communication, as these engineered features and devices are necessary for a secure system.

3. **Incorporate engineered features or devices to control the potential for loss.**

   If preventing, limiting, or reducing the potential for loss through design alteration and material selection is not feasible or adequate, employ engineered features and devices to control loss associated with susceptibility, hazard, and vulnerability. In general, engineered features actively disrupt the loss scenario sequence and interactions, and devices reduce the potential, severity, and extent of loss.

   There are two general types of engineered features and devices employed to address the potential for loss associated with the *system function of interest*:

- *Mandatory security features and devices:* Mandatory security features and devices are those that apply foundational security principles for the interfaces. For example, each interface must have mediated access to control access to and use of the capability and data provided by the interface.
- *Function-specific features and devices:* Function-specific security features and devices protect against a loss associated with the design's ability to meet functional requirements and performance parameters. Engineered features such as redundant data and control flows and redundant system elements can supplement the design selection to achieve the required protection. The system may also have engineered features that enable external entities to intervene into the system to address the potential, severity, or extent of loss.

4. **Provide visibility and feedback to external entities.**

If design alteration, material selection, and engineered features and devices are not feasible or do not adequately lower the frequency, potential, severity, or extent of loss caused by the susceptibility, hazard, or vulnerability, employ engineered detection and feedback systems and warning devices to alert external entities to the presence of a susceptible, hazardous, or vulnerable condition; the occurrence of an event that will lead to a loss; or an actual loss event. External entities include operational personnel, monitoring systems, or other systems capable of responding.

Example: Engineered anomaly detection features can be used to provide situational awareness data and warnings to system users.

Note: The visibility provided is not of value if the external entities are not able to respond appropriately. For example, personnel should have appropriate training and standard operating procedures for loss.

5. **Incorporate signage, procedures, training, and proper equipment**.

Incorporate procedures, training, signage, and proper equipment where design alternatives, design changes, and engineered features and devices are not feasible and warning devices cannot adequately lessen the potential, severity, or extent of loss caused by the hazard, susceptibility, or vulnerability. Procedures and training include appropriate warnings and cautions and may prescribe the use of equipment. For critical losses, the use of signage, procedures, training, and equipment as the only means to reduce the potential, severity, or extent of loss should be avoided.

Example: Procedures and training materials address proper use of the *system function of interest*, as well as the use of mediated access functions, redundant capabilities, and warning systems, including all relevant cautions and warnings.

TRUSTWORTHY SECURE DESIGN

Trustworthy secure design is a means to optimally satisfy the requirements that form the basis for achieving system security objectives across competing and conflicting stakeholder capability needs, concerns, and constraints.

D.4 FUNCTIONAL DESIGN CONSIDERATIONS

This section describes the functional design considerations for trustworthy secure systems. These include assured functions that provide control enforcement, control decision, and control infrastructure; the design criteria for mechanisms; security function failure analysis; and trade space considerations.

D.4.1 Roles for Security-Relevant Control

Historically, from the perspective of secure system design and evaluation, the term *security relevance* has been used to differentiate the role of system functions that singularly or in combination exhibit behavior, produce an outcome, or provide a capability to enforce authorized and intended system behavior or outcomes. This includes those authorized behaviors and outcomes associated with protective failure and protective recovery in the event of loss. However, from the perspective of the views of security (Section 2.3.8) and the possibility of loss due to weaknesses and defects in any system function, all functions have loss- related concerns and, thus, protection concerns. The active protection functions enforce or contribute to the control or influence of the behaviors and outcomes of the system or system elements, and all functions have the potential to influence behaviors and outcomes beyond themselves and their host system elements. Therefore, protection control functions may be characterized and analyzed by using the following designations:

- **Protection Control Decision Functions:** These functions make authorization decisions or take other actions for protection control enforcement functions. For example, a protection control decision function is a function that decides to grant or deny access to a resource based on a request, possibly from a protection control enforcement function.
- **Protection Control Enforcement Functions:** These functions enforce a constraint to ensure that the system or system element exhibits only authorized and intended behaviors or outcomes. For example, a protection control enforcement function enforces a decision to grant or deny access to a resource.
- **Protection Control Infrastructure Functions:** These functions support and help protection control enforcement and control decision functions fulfill their purposes. The functions also provide data or services or perform operations upon which protection control enforcement and decision functions depend. For example, a protection control infrastructure function includes secure storage, secure communication, and anomaly detection mechanisms.

Other functions, some of which may be control functions for other purposes besides protection, can potentially adversely affect the correct operation of the protection control enforcement, decision, and infrastructure functions. For the purposes of secure design and evaluation, these functions are designated *other system functions*. Ideally, these functions should be non-interfering functions. The objective for non-interference may be achieved through assurance with constraints on the requirements, architecture, design, and use of these functions.

All system functions can be mapped to one or more of the functions listed above for the purpose of secure design and evaluation. The importance of the distinction is to guide and inform a principled design to limit interference among functions with confidence. Such confidence can be achieved by employing *Trustworthy System Control*, applying the design criteria described in Section D.4.2, and optimally placing a function in the system architecture to limit the side effects and interactions that may interfere with the protection control decision, protection control enforcement, and control infrastructure functions.

System analyses can also determine the extent to which functions may interfere with other functions and inform uncertainty that impacts confidence and needed actions for assurance. For example, to satisfy a size or form-factor constraint, a system function may occupy the same privilege domain as control enforcement, control decision, or control infrastructure functions, thereby elevating the privilege of that system function. If the size or form-factor constraint does not exist, it would be prudent to employ that system function elsewhere to avoid giving the function elevated privilege. This would increase the assurance that the enforcement, decision, and infrastructure functions are isolated from the other parts of the system and would not be adversely impacted by their behavior or provide an avenue for attack.

D.4.2 Essential Design Criteria for Mechanisms

To effectively achieve the objectives of trustworthy secure design, engineered features and devices – often known as *mechanisms* – must satisfy four essential design criteria. They must be non-bypassable, evaluatable, always invoked, and tamper-proof [Uchenick05]. In general, the design for any control function that provides protection should adhere to those criteria.[80] A brief description of the essential design criteria is provided in Table D-1.

TABLE D-1: ESSENTIAL DESIGN CRITERIA FOR MECHANISMS

ESSENTIAL DESIGN CRITERIA	DESCRIPTION
NON-BYPASSABLE	The mechanism must not be circumventable.
EVALUATABLE	The mechanism must be sufficiently small and simple enough to be assessed to produce adequate confidence in the protection provided, the constraint (or control objective) enforced, and the correct implementation of the mechanism. The assessment includes the analysis and testing needed.
ALWAYS INVOLKED	The protection provided by a mechanism or feature that is not always invoked is not continuous and therefore, a loss may occur while the mechanism or feature is suspended or turned off.
TAMPER-PROOF	The mechanism or feature and the data that the mechanism or feature depends on cannot be modified in an unauthorized manner.

The design criteria described above are based on the *generalized reference monitor concept*. The reference monitor concept[81] is an abstract model of the necessary and sufficient properties that must be achieved by any mechanism that performs an access mediation control function [Levin07] [Anderson72]. The reference monitor concept is a foundational access control concept for assured system design. It is defined as a trustworthy abstract machine that mediates all

[80] The argument that any control function should be non-bypassable, evaluatable, always invoked, and tamper-proof follows from an in-depth examination of Systems Theoretic Process Analysis (STPA) as described in [Leveson11], specifically the discussions on why controls may fail and how to address failure.

[81] The *reference monitor concept* is described in the *Trustworthy System Control* principle in Appendix E.

accesses to objects by subjects [TCSEC85]. As a concept for an abstract machine, the reference monitor does not address any specific implementation. A reference validation mechanism, which includes a combination of hardware and software, realizes the reference monitor concept to provide the access mediation foundation for a trustworthy secure system.

The generalized reference monitor concept and the four essential design criteria can be used effectively as the design basis for individual system elements, collections of elements, networks, and systems where intentional and unintentional adversity can prevent the realization of a loss control objective. The reference monitor concept also drives the need for rigor in engineering activities commensurate with the trust to be placed in the system or its constituent system elements.[82] The concept describes an *abstract model* of the necessary properties that must be realized by any mechanism that claims to achieve a constraint or set of constraints and the basis for determining the extent to which the properties are satisfied. A mechanism that achieves successful constraint has two parts: (1) a means to decide whether to constrain or not constrain, and (2) the enforcement of the decision. Enforcement of the decision must sufficiently:

- Enforce constraints to achieve only the authorized and intended system behaviors and outcomes
- Provide self-protection against targeted attacks on the mechanism enforcing the decision (including the application of the essential design criteria)
- Be absent of self-induced emergent, erroneous, unsafe, and non-assured control actions

The protection characteristics for mechanisms must account for but not be dependent on having detailed knowledge of the capability, means, and methods of an adversary.

THE SCIENCE BEHIND THE SECURITY

"Each of these [design] requirements [for mechanisms] is significant, for without them, the mechanism cannot be considered secure. The [need to be tamper-proof] is obvious, since if the reference validation mechanism can be tampered with, its validity is destroyed, as is any hope of achieving security through it. The [third] requirement of always invoking the reference validation mechanism simply states that if the reference validation is (or must be) suspended for some group of programs, then those programs must be considered part of the security apparatus and be [tamper-proof and evaluatable]. The [evaluatable] requirement is equally important. It states that because the reference validation mechanism is the security mechanism in the system, it must be possible to ascertain that it works correctly in all cases and is always invoked. If this cannot be achieved, then there is no way to know that the reference validation correctly takes place in all cases, and therefore there is no basis for certifying a system as secure."

-- James P. Anderson
***The Anderson Report* [Anderson72]**

D.4.3 Security Function Failure Analysis

The design principle of *Protective Failure* states that a failure of a particular system element should neither result in an unacceptable loss nor invoke another loss scenario. The failure of a

[82] Conceptually, the reference monitor concept can be extended to any control function that is to enforce a system constraint [MITRE21].

security function is of special concern, given the need for security functions to always be invoked and operating correctly. Consequently, failure analyses must be performed during system design to determine the impacts of function failure on the system capabilities, including the protection capability relative to the resulting consequences of such failure and the needed assurance of the protection capability.

Failure analyses consider the assets that may be impacted by security function failure and the associated loss consequences. Failure analyses also consider the function allocation to system elements and the way the system function and element combination interacts with other system function and element combinations, independent of specific events and conditions that might lead to the failure. The principles for trustworthy secure design in Appendix E serve to guide and inform the analyses.

The outcomes of the security function failure analyses also drive assurance levels and objectives, as well as the fidelity and rigor of architecture, design, and implementation methods employed to achieve those objectives. Assurance considerations are discussed in Appendix F.

D.4.4 Trade Space Considerations

System design involves a number of trade space decisions. These decisions may be informed by criticality or priority of an asset, costs, and benefits of an approach. Decision-making about protecting the various assets includes determining the criticality (e.g., assessing the positive effect in achieving objectives and the negative effect if there is some loss associated with the asset) and priority (i.e., relative ranking of equally critical assets) of each asset. The criticality and priority based on *valuation* are used in investment decisions on the type, rigor, and expected effectiveness of protection.

The *costs* associated with a trustworthy secure design approach include the cost to acquire, develop, integrate, operate, and sustain the security features; the cost of the security features and functions in terms of their system performance impact; the cost of security services used by the system; the cost of developing and managing life cycle documentation and training; and the cost of obtaining and maintaining the target level of assurance.

The cost of analysis to substantiate the trustworthiness claims of certain design choices is also an important trade space factor. Given two equally effective design options, the more attractive of the two options may be the one that has a lower relative cost to obtain the assurance needed to demonstrate satisfaction of trustworthiness claims. In all cases, the cost of system security must be assessed at the system level and consider trustworthiness objectives and the cost that is driven by the assurance activities necessary to achieve the trustworthiness objectives. Trustworthiness design principles such as *Commensurate Rigor* and *Commensurate Trustworthiness* inform the trade space analysis.

The benefits derived from a trustworthy secure design approach are determined by its effectiveness in providing the required protection capability, the trustworthiness that can be placed on it, and the loss potential associated with it, given the value, criticality, exposure, and importance of the assets protected. It may be the case that an *optimal balance* between cost and benefit is realized through the use of a less costly combination of engineering activities and system features and functions rather than the use of a single cost-prohibitive activity or security feature or function. It may also be the case that the adverse performance impact on the system may preclude some security options.

> *"Retroactive cybersecurity design is a Sisyphean task."*
>
> -- **O. Sami Saydjari**
> ***Engineering Trustworthy Systems*** **[Saydjari18]**

APPENDIX E

PRINCIPLES FOR TRUSTWORTHY SECURE DESIGN

FOUNDATIONS FOR ENGINEERING TRUSTWORTHY SECURE SYSTEMS[83]

This section describes a set of principles that serve as the foundation for engineering trustworthy secure systems. The principles for trustworthy secure design are applied to control the adversity[84] that might occur as a direct or indirect result of the system delivering a specified capability at a specified level of performance. The principles represent research, development, and application experience starting with the early incorporation of security mechanisms for trusted operating systems to today's fully networked, distributed, mobile, and virtual computing components, environments, and systems. The principles are intended to be universally applicable across this broad range of systems, as well as new systems as they emerge and mature.

The principles for trustworthy secure design provide a basis for reasoning about a system. As reasoning tools, the inherent suitability of the principles in a particular situation will depend on the judgment of the practitioner. Engineering judgment must be exercised in the application of the principles for trustworthy secure systems.[85] The principles should not be applied as "rules" to be complied with, nor should they be prioritized, sequenced, or ordered for prescriptive application, or used individually or in groups as a basis for making judgments of conformance. Principles are subject to various priorities and constraints that may restrict or preclude their application. At times, these principles may be in conflict with other principles and must be deconflicted. In practice, the principles can be satisfied or implemented in various and perhaps equally effective ways. Within the system life cycle, the applicability of a particular principle may change due to evolving requirements, protection needs, priorities, or constraints; architecture and design decisions and trade-offs; or changes in the risk acceptance threshold.

KEY SECURITY OBJECTIVE

An important objective for security is the reduction in uncertainty regarding the occurrence and effects of adverse events. Reducing the uncertainty of adverse events is achieved by eliminating hazards, susceptibility, and vulnerability to the extent possible. Where elimination cannot occur, their effects must be controlled. Applying the design principles for trustworthy secure systems is a part of the means to achieve both the elimination and the control of the hazards, susceptibility, and vulnerability that lead to adverse events [MITRE21].

---

[83] NIST acknowledges and appreciates the contributions of the Naval Postgraduate School Center for Information Systems Security Studies and Research and The MITRE Corporation in providing content for this appendix. The content was guided and informed by the research reports of the principal investigators from those organizations [Levin07] [MITRE21].

[84] The term *adversity* refers to the conditions that can cause a loss of assets (e.g., threats, attacks, vulnerabilities, hazards, disruptions, and exposures).

[85] Engineering judgment considerations for the application of the principles for trustworthy secure systems is described in [MITRE21].

The principles for trustworthy secure design are representative of the practices of the safety, security, survivability, and resilience communities and the specialty engineering disciplines associated with those communities. Collectively, the goals of these practices represent the "end objectives" that the system must satisfy for trustworthy control of adverse effects. The concepts and theorems from the disciplines of computer science, systems engineering, control systems, fault/failure tolerance, software engineering, computer engineering, and mathematics – as employed across the communities and specialties – constitute the means to achieve the end objectives. The application of the principles should be planned for, appropriately scoped, and revisited throughout the system life cycle and engineering effort.

The principles for trustworthy secure design are listed in Table E-1. The principles are divided into two categories: (1) *trustworthiness* design principles, and (2) *loss control* design principles.

TABLE E-1: PRINCIPLES FOR TRUSTWORTHY SECURE DESIGN

PRINCIPLES FOR TRUSTWORTHY SECURE DESIGN	
TRUSTWORTHINESS DESIGN PRINCIPLES	**LOSS CONTROL DESIGN PRINCIPLES (Cont.)**
Clear Abstractions	Defense In Depth
Commensurate Rigor	Distributed Privilege
Commensurate Trustworthiness	Diversity (Dynamicity)
Compositional Trustworthiness	Domain Separation
Hierarchical Protection	Least Functionality
Minimized Trusted Elements	Least Persistence
Reduced Complexity	Least Privilege
Self-Reliant Trustworthiness	Least Sharing
Structured Composition and Decomposition	Loss Margins
Substantiated Trustworthiness	Mediated Access
Trustworthy System Control	Minimize Detectability
LOSS CONTROL DESIGN PRINCIPLES	Protective Defaults
Anomaly Detection	Protective Failure
Commensurate Protection	Protective Recovery
Commensurate Response	Redundancy
Continuous Protection	

E.1 TRUSTWORTHINESS DESIGN PRINCIPLES

Trustworthiness design principles are based on the historical meaning of trustworthiness and trust and their use as the basis for the design of secure systems. In particular, [Neumann04] defines the terms *trustworthiness* and *trust* as follows:

- **Trustworthiness:** The demonstrated worthiness of an entity to be trusted based on evidence that supports a claim or judgment of being trustworthy.
- **Trust:** A belief that an entity *can* be trusted. (Implies that trust may be granted to an entity whether the entity is trustworthy or not).

Trustworthiness is a cross-cutting objective in the design of systems due to the consequences of the failure of systems to behave and produce outcomes only as authorized and intended. The terms *trust* and *trusted* are used to mean "the decision is made to trust because the required trustworthiness is demonstrated." Trustworthiness is associated with one of the essential design criteria and the reference monitor concept (Section D.4.2). A protection mechanism or feature must be evaluatable (i.e., the mechanism must be sufficiently small and simple enough to be assessed to produce adequate confidence in the protection provided, the constraint or control objective enforced, and the correct implementation of the mechanism).

Trustworthiness design principles are fundamental to managing complexity and otherwise aid in understanding the engineered system. The principles are necessary to achieve loss control objectives given the complexity in understanding loss in context (based on how the system is intended to be utilized and sustained). Complexity increases analysis workloads and reduces confidence in that analysis. Complexity also increases the costs and difficulty of performing systems analyses for loss. That is, systems may be too complex to be analyzed for adequate assurance [Sheard18].

The trustworthiness design principles include:

- Clear Abstractions
- Commensurate Rigor
- Commensurate Trustworthiness
- Compositional Trustworthiness
- Hierarchical Protection
- Minimized Trusted Elements
- Reduced Complexity
- Self-Reliant Trustworthiness
- Structured Decomposition and Composition
- Substantiated Trustworthiness
- Trustworthy System Control

E.1.1 Clear Abstractions

PRINCIPLE: *The abstractions used to characterize the system are simple, well-defined, accurate, precise, necessary, and sufficient.*

Note: Abstractions can help manage the complexity of the system [ISO 24765]. Clarity in the abstract representations of the system helps to facilitate an accurate understanding of the system and how the system functions to deliver the required capability. Clear abstractions also reduce the potential for misunderstanding or misinterpretation of what is represented by the abstraction. Applying the principle of clear abstractions means that a system has simple, well-defined interfaces and functions that provide a consistent and intuitive view of the data and how it is managed. The elegance (e.g., accuracy, precision, simplicity, necessity, sufficiency) of the system interfaces – combined with a precise definition of the functional behavior of the interfaces – promotes ease of analysis, inspection, and testing, as well as the correct and secure use of the system. Examples that reflect the application of this principle include avoidance of redundant, unused interfaces; information hiding;[86] and avoidance of semantic overloading of interfaces or their parameters (e.g., not using one function to provide different functionality, depending on how it is used).

It is important to ensure that the appropriate rigor is applied in the development of system abstractions during design. Clarity in the abstract representation of the system requires the use of well-defined syntax and semantics with elaboration as needed to ensure the representations are well-defined, precise, necessary, and sufficient. Clear abstractions promote confidence in analysis, verification, and the correct use of the system. Abstractions can be achieved through the use of models, including Systems Modeling Languages.

REFERENCES: [ISO 24765]; [Schroeder77]; [Neumann04]; [Levin07].

E.1.2 Commensurate Rigor

PRINCIPLE: *The rigor associated with the conduct of an engineering activity provides the confidence required to address the most significant adverse effect that can occur.*

Note: Rigor determines the scope, depth, and detail of an engineering activity. Rigor is a means to provide confidence in the results of a completed engineering activity. Generally, an increase in rigor translates into an increase in confidence in the results of the activity. Further, increased confidence reduces the uncertainty that can also reduce risk or provide a better understanding of what to address to achieve risk reduction. The relationship between rigor and the criticality of data and information used to make decisions is recognized by systems analysis practice [ISO 15288].

The principle of commensurate rigor helps to ensure that the concept of rigor is included as an equal factor in the trade space of capability, adverse effect, cost, and schedule in the planning and conduct of engineering activities, method and tool selection, and personnel selection. An increase in rigor may translate into an increase in the cost of personnel, methods, and tools required to complete rigorous engineering activities or an increase in schedule to accomplish the activities with the expected rigor. Any increased cost that may occur can be justified by acquiring confidence about system performance to limit loss while also addressing the system's ability to deliver the capability. Therefore, the rigor associated with an engineering activity should be commensurate to the significance of the most adverse effect associated with the activity.

REFERENCES: [ISO 15288]; [Neumann04].

E.1.3 Commensurate Trustworthiness

PRINCIPLE: *A system element is trustworthy to a level commensurate with the most significant adverse effect that results from a failure of that element.*

Note: A trusted element continuously exhibits properties of trust for the duration of the time that it is depended upon by other system elements. The degree of trustworthiness needed for a trusted element is determined by those entities that depend on the element. Some basis is

[86] The term *information hiding*, also called representation-independent programming, is a design discipline to ensure that the internal representation of information in one system component is not visible to another system component invoking or calling the first component, such that the published abstraction is not influenced by how the data may be managed internally.

required to support decisions about trust and trustworthiness. The basis includes expressing the trust that is to be placed in a system element, expressing the trustworthiness that is exhibited by the element, and comparing the trustworthiness of different system elements. This principle is particularly relevant when considering systems and elements in which there are complex chains of trust dependencies.

REFERENCES: [Schroeder77]; [Neumann04].

E.1.4 Compositional Trustworthiness

PRINCIPLE: *The system design is trustworthy for each aggregate composition of interacting system elements.*

Note: The trustworthiness of an aggregate of composed system elements cannot be assumed based on the trustworthiness assertions of each element in the aggregate. Further, the trustworthiness of an aggregate of composed trustworthy system elements cannot be assumed to be equal to the trustworthiness of the least trustworthy element in the aggregate. By definition, a system is a combination of interacting system elements. Each system function results from the emergent behavior of a composed set of system elements. Likewise, the trustworthiness of a composed set of elements is an emergent property of the composition. Therefore, the trustworthiness of the composed set of system elements (i.e., aggregate) for a given system function must be determined by treating the aggregate as a single discrete element. The compositional trustworthiness principle addresses how an argument can be made for system-level trustworthiness given how the constituent elements of the system compose to form the system and do so by adhering to the composition principles.

REFERENCES: [ISO 15288]; [Neumann00]; [Neumann04]; [Leveson11].

E.1.5 Hierarchical Protection

PRINCIPLE: *A system element need not be protected from more trustworthy elements.*

Note: Hierarchical protection is a simplifying assumption for trade decisions to help determine where emphasis is placed in providing protection and the extent of the protection effectiveness. The simplifying assumption introduces susceptibilities to system elements that are dependent on more trustworthy elements. The assumption relies on validated trust assertions about the more trustworthy element and acceptable uncertainty associated with behavior outside of the scope of the validated trust assertions. For example, systems may include a human element, which is often the more trustworthy element. The assertions of the trusted human are violated for the malicious insider threat. The extent to which any element is considered trustworthy has limits, and beyond those limits, the element should not be assumed to remain trustworthy. In the degenerate case of the most trustworthy system element, it must protect itself from all other elements. For example, if an operating system kernel is deemed the most trustworthy component in a system, then it must protect itself from the less trustworthy applications it supports. However, the applications do not need to protect themselves from the operating system kernel.

REFERENCES: [Neumann04]; [Smith12]

E.1.6 Minimized Trusted Elements

PRINCIPLE: *A system has as few trusted system elements as practicable.*

Note: Minimizing trusted system elements is a cost-benefit trade space consideration employed for the functional allocation of trust within the system. The need for trust is tied to the function provided by a system element, and that need is independent of any distribution of trust across multiple elements in the architecture. The trade decision is, therefore, how best to allocate trust to system elements given the functions they provide and how the elements are best distributed throughout the architecture where there is justified need for the distribution. The minimization of trusted system elements is one of several considerations in making that decision.

Trusted elements are generally costlier to construct due to increased rigor in engineering processes and activities. They also require more analysis to qualify their trustworthiness. Minimizing the number of trusted system elements reduces the cost of analysis (i.e., decreases the size, scope, and complexity of the analysis). When the minimization of trusted system elements considers the principle of *Commensurate Protection*, the cost-effectiveness of the analysis is also ensured (i.e., cost of the analysis is justified by the extent of trust required).

Historically, the analysis of interactions between trusted system elements and untrusted system elements is one of the most important aspects of the trust-based verification of system security performance. If these interactions are unnecessarily complex, the security of the system will also be more difficult to ascertain than one whose internal trust relationships are simple and elegantly constructed. In general, fewer trusted components will result in fewer internal trust relationships and a simpler system.

REFERENCES: [Schroeder77]; [Neumann04]; [Smith12]; [Saltzer09].

E.1.7 Reduced Complexity

PRINCIPLE: *The system design is as simple as practicable.*

Note: Many engineered systems are complex. Complexity can be found in the system structure, interfaces, dependencies, data and control flows, and the system's interaction with its external environment. Some degree of complexity in the system design is inherent, unavoidable, and must be accepted. The objective is to ensure that the design reflects the extent to which complexity can be reasonably minimized (i.e., avoid unnecessary complexity). Simplicity in the system design reduces complexity, allows for increased confidence in the ability to understand the design, and is less prone to error. A simpler design is less prone to erroneous interpretation during system analysis, system implementation, and system verification [Moller08]. Reduced complexity contributes to confidence in the technical understanding of the design, enabling more informed trade decisions. It also facilitates the identification of vulnerabilities and the verification of the correctness and completeness of system security functions.

Complexity is impacted by how the system is decomposed into constituent elements, aggregates of elements (e.g., subsystems, assemblies), and the composition of those elements to comprise the system. Identifying and assessing loss scenarios, susceptibilities, and vulnerabilities is made more difficult by complexity. Thus, reducing complexity helps to facilitate the identification and assessment of loss scenarios, hazards, susceptibility, and vulnerability to all forms of adversity. Finally, any conclusion about the correctness, completeness, and existence of vulnerabilities in systems or system elements can be reached with a higher degree of assurance in contrast to conclusions reached in situations where the system design is inherently more complex. The principle of reduced complexity may also be referred to as the principle of simplification or least common mechanism.

REFERENCES: [Saltzer75]; [Neumann04]; [Jackson13]; [Saleh14]; [Moller08].

E.1.8 Self-Reliant Trustworthiness

PRINCIPLE: *The trustworthiness of a system element is achieved with minimal dependence on other elements.*

Note: In the ideal case, the trustworthiness of a system element occurs when the claim of trustworthiness is not dependent on protection from another element. If an element is dependent on some other element to satisfy its trustworthiness claims, then that element's trustworthiness is susceptible to any loss or degradation of the protection capability provided by the other element. The considerations for the extent to which a system element exhibits self-reliant trustworthiness include:

- The trustworthiness objective for the capability
- The trustworthiness of the system element in providing the capability
- The extent to which the capability provided by a system element is dependent on another element
- The extent to which the trustworthiness associated with a capability is dependent on another system element

An argument for self-reliant trustworthiness can be applied at the discrete system element level, at the level of an aggregate of elements, at the system level, or at the system of systems level. In all cases, the distinction between the capability provided and the trustworthiness responsibility for that capability must be preserved (e.g., self-reliant trustworthiness cannot be claimed if the protection assertions for trust are allocated to and therefore dependent on some other entity). Likewise, when a capability is distributed across multiple system elements, self-reliant trustworthiness requires that the trust expectations for the capability are properly allocated across the elements that comprise the distributed capability.

The judgment that a system element is self-reliantly trustworthy is based on the element's ability to satisfy a specific set of requirements and associated assumptions. An element that is self-reliantly trustworthy for one set of requirements and assumptions is not necessarily self-reliantly trustworthy for other sets of requirements and assumptions. Any change in the requirement, the satisfaction of the requirement, or in the assumptions associated with the requirement requires reassessment to determine that the element remains self-reliantly trustworthy.

REFERENCES: [Neumann04].

> *"System components [elements] are self-protective. System componentry is augmented, upgraded, and replaced over time by methods and personnel that cannot be unequivocally trusted."*
>
> -- **An Objective of the Security in the Future of Systems Engineering** [FUSE21]

E.1.9 Structured Decomposition and Composition

PRINCIPLE: *System complexity is managed through the structured decomposition of the system and the structured composition of the constituent elements to deliver the required capability.*

Note: The structured decomposition of the system and the subsequent composition of the constituent system elements are guided and informed by the concepts of modularity, layering, and partially ordered dependencies. Modularity is the system design technique to "divide and conquer" – that is, sub-divide the system into smaller, well-defined cohesive components and assemblies that are referred to as modules. Modularity serves to isolate functions and data structures into well-defined logical units. Modular decomposition can include the allocation of policies to systems in a network, the allocation of system policies to layers, the separation of system applications into processes with distinct address spaces, and the separation of processes into subjects with distinct privileges based on hardware-supported privilege domains. Modular design may also extend to consider trust, trustworthiness, privilege, and policy.

Layering is the grouping of modules into a relational structure with well-defined interfaces, function, data, and control flow so that the dependencies graph among layers is linearly or partially ordered such that higher layers are dependent only on lower layers [Neumann04]. Partially ordered dependencies among modules (e.g., if module A depends on module B, then module B cannot depend on module A) and system layering contribute significantly to system design simplicity and coherence. While a partial ordering of all functions and processes may not be possible, the inherent problems of circularity can be more easily managed if the circular dependencies are constrained to occur within layers and minimized within each layer. Partially ordered dependencies also facilitate system testing and analysis and enable a strong form of loose coupling (i.e., minimizing interdependencies among modules).

Modularity and layering are effective in managing the complexity of the composed system. They provide the means to decompose the system into discrete and aggregate elements to better comprehend the system in terms of its structure, flows, relationships, and how the system delivers the required capability. The structured composition of the constituent elements must also adhere to the principle of *Compositional Trustworthiness* to provide a basis to support claims about how the system is composed based on the application of modularity, layering, and partially ordered dependencies to achieve authorized and intended behaviors and outcomes.

REFERENCES: [Saltzer75]; [Schroeder77]; [Neumann04]; [Simovici08]; [Adcock20].

E.1.10 Substantiated Trustworthiness

PRINCIPLE: *System trustworthiness judgments are based on evidence that demonstrates the criteria for trustworthiness have been satisfied.*

Note: Trustworthiness should not be assumed but rather substantiated through evidence that clearly enables determination of the extent to which an entity is worth being trusted. This helps to ensure that an entity is never trusted beyond the extent to which it is worthy of trust. The approach to substantiated trustworthiness requires commensurate rigor with cautious mistrust (i.e., system elements are assumed to be guilty until they are proven innocent).[87] Substantiated trustworthiness is characterized by a design mentality in which all components involved in the

[87] Adapted from a statement made by John Rushby, SRI International, about the need for software to be treated as "guilty until proven innocent" at a Layered Assurance Workshop (LAW).

design context (i.e., a system element and the elements with which it interacts) are treated with a mutually suspicious mindset [Schroeder77][Neumann04]. Such mutual suspicion reflects cautious distrust – the feeling or thought that something undesired, unwanted, or unexpected is possible or can happen. The design for every system element should reflect a lack of trust in interacting elements or itself. This suspicion assumes element non-performance and addresses the following two cases:

- **Interacting element suspicion (mutual suspicion):** The design for the system element-of-interest is based on the non-performance of the elements it interacts with and how their non-performance can influence the behavior and outcomes produced by the element-of-interest. Mutual suspicion may also be referred to as zero trust.[88] Designing to mutual suspicion is reinforced by applying the principle of *Least Privilege* to all entities (so an element executes with only the privileges needed, mitigating harm that may be created) while applying the principle of *Least Persistence* so that each element is minimally exposed.
- **Self-suspicion:** The design for the system element-of-interest must consider its own non-performance independent of any external influence. Designing to self-suspicion may involve self-monitoring and built-in actions, including built-in testing at the initiation of the element.

This approach forces the system designer to assume things will not go right and to rigorously seek evidence that demonstrates the effectiveness of the design when things go wrong.

Considerations for system element non-performance include:

- The expectation that design elements will behave and produce outcomes that are inconsistent with their design intent
- The constraints, assumptions, and preconditions associated with achieving threshold performance
- Intentional and unintentional events and conditions, typically referred to by terms like fault, error, failure, and compromise

REFERENCES: [Neumann04]; [Levin07]; [Schroeder72].

E.1.11 Trustworthy System Control

PRINCIPLE: *The design for system control functions conforms to the properties of the generalized reference monitor.*

Note: The trustworthy system control principle reflects the generalization of the reference monitor concept to provide a uniform design assurance basis for trustworthy system control mechanisms or constraint-enforcing mechanisms that compose to provide system control functions. The reference monitor concept (Section D.4.2) is a foundational access control concept for secure system design. It is defined as a trustworthy abstract machine that mediates all accesses to objects by subjects [TCSEC85]. As a concept for an abstract machine, the reference monitor does not address any specific implementation. A reference validation mechanism, a combination of hardware and software, realizes the reference monitor concept to provide the access mediation foundation for a secure system [Anderson72].

[88] *Zero trust* means only that an entity is not trusted; zero trust does not mean that the entity is not trustworthy. The term *zero trust* is not to be confused with Zero Trust Architecture (ZTA).

The reference monitor concept has three criteria that provide design assurance of its realization as a reference validation mechanism:

- The reference validation mechanism must be tamper-proof, ensuring that its integrity and validity is not destroyed.
- The reference validation mechanism must always be invoked, and if it cannot be, then the group of programs for which it provides validation services must be considered part of the reference validation mechanism and be subject to the first and third requirements.
- The reference validation mechanism must be subject to rigorous analysis and tests, the completeness of which can be assured (with the purpose of ascertaining that the reference validation mechanism works correctly in all cases).

For trustworthy system control, a fourth criterion of non-bypassability is added (Section D.4.2). Successful achievement of the criterion will prevent the interference of outside entities on a protection mechanism or controller. More specifically:

- A protection mechanism or feature should not be circumventable (i.e., the mechanism should be non-bypassable).
- A protection mechanism or feature should be evaluatable (i.e., sufficiently small and simple enough to be assessed to produce adequate confidence in the protection provided, the constraint or control objective enforced, and the correct implementation of the mechanism [see *Reduced Complexity*]).
- A protection mechanism or feature is always invoked, providing continuous protection.
- A protection mechanism or feature must be tamper-proof (i.e., neither the protection functions nor the data that the functions depend on can be modified without authorization).

Trustworthy system control also uses *protective control*. Protective control encompasses control, safety, and security concepts to establish a system capability that sufficiently:

- Enforces constraints to achieve only the authorized and intended system behaviors and outcomes
- Provides self-protection against targeted attack on the system
- Is absent of self-induced emergent, erroneous, unsafe, and non-secure control actions

The notion of protective control underlies the loss control objectives and transforms the approach for design to not be dependent on having detailed knowledge of the capability, means, and methods of an adversary. This design approach can be employed in attack-dependent or attack-independent manners based on the limits of certainty for what is known with confidence about the adversary.

Trustworthy system control serves well as the design basis for individual system elements, collections of elements, networks, and systems where intentional and unintentional adversity can prevent the achievement of the loss control objectives. The principle also drives the need for rigor in engineering activities commensurate to the trust placed in the system elements.

REFERENCES: [Levin07]; [Anderson72]; [TCSEC85]; [Uchenick05].

E.2 LOSS CONTROL DESIGN PRINCIPLES

Loss control design principles are applied in combination with the trustworthiness principles to yield trustworthy control over the system behavior and outcomes, deliver the required system capability, and protect against loss. The loss control design principles include:

- Anomaly Detection
- Commensurate Protection
- Commensurate Response
- Continuous Protection
- Defense In Depth
- Distributed Privilege
- Diversity (Dynamicity)
- Domain Separation
- Least Functionality
- Least Persistence
- Least Privilege
- Least Sharing
- Loss Margins
- Mediated Access
- Minimize Detectability
- Protective Defaults
- Protective Failure
- Protective Recovery
- Redundancy

E.2.1 Anomaly Detection

PRINCIPLE: *Any salient anomaly in the system or in its environment is detected in a timely manner that enables effective response action.*

Note: The purpose of anomaly detection is to identify the need to take corrective action to address a loss condition that has occurred or that will occur if conditions that affect the system behavior are allowed to persist. Anomaly detection is critical to achieving the loss control objectives to prevent and limit loss and its adverse effects. The detection of such anomalies requires monitoring system behaviors and outcomes to confirm that they have not deviated from the design intent. It also requires monitoring conditions in the environment to identify or forecast those conditions that can cause an anomaly in the system if corrective action is not taken. The "timely manner" aspect of anomaly detection reflects the urgency to detect emerging loss conditions as early as possible. Early detection increases response action options, such as graduated response options, and ensures that response actions have sufficient time to have an effect. When the determination of response involves humans in the loop, early detection enables a more reasoned judgment of appropriate response.

Anomaly detection can be implemented at varying levels of abstraction (e.g., system, subsystem, assembly, function, mechanism) and may occur in periodic, aperiodic, or event-driven manners. The basis for anomaly detection within the system is the expectation that the system behaviors, outcomes, and interactions produced are expected to remain consistent, adhere to some norm, or are deterministic across all system states and modes. The types of anomalies include those associated with the results of system behavior; state consistency; continuity of function; integrity, correctness, and trustworthiness of system elements; system configuration; and the abuse or misuse of the system.

The basis for anomaly detection in the environment differs from that in the system because the environment is not within the control of the system. The environment presents a wide range of adversity to the system, and the system is designed to achieve its design intent within defined bounds of environmental conditions. Those bounds can be treated as the “norm” for anomaly detection, whereby environmental conditions that are trending beyond the norm or that reflect conditions outside of the norm may result in an adverse effect on the system, thus requiring a planned response to prepare for an impending difficulty or crisis.

Anomaly detection requires capturing data to support all intended response actions for a detected anomaly, including attribution-related data. Consequently, the rigor in data describing the anomaly must be commensurate with the consequences of the loss scenarios associated with the anomaly and of wrong responses in addressing the detected anomaly. The responses taken will often rely on attribution to uniquely identifiable entities that may be responsible for undesired actions, behaviors, or outcomes. For non-human entities, corrective actions may include component replacements, repairs, or other corrections. For human entities, these may include training, remediation, or disciplinary actions. Wrongful attribution may have undesired consequences, such as the cost of unnecessarily repairing the wrong system element while an undesired condition persists or the wrongful termination of an individual. Attribution rigor is driven by the needed proof that an entity is responsible for an anomaly. Three aspects of anomaly detection are necessary to provide criteria for an appropriate response action or set of actions:

- **Basis for Correctness:** A system model provides a basis against which actual behavior and outcomes can be compared to confidently enable conclusions that an anomaly exists or to determine or forecast that an anomaly is about to occur. System models includes normal, contingency, degraded, and other system states/modes of operation and account for the adversity to which the system is subjected.

- **Data Collection:** Systems capture self-awareness data in the form of health, status, test, and other data indicative of actual behavior and outcomes, including traceability to support attribution. Terms for data collection include instrumentation, monitoring, logging, auditing, self-tests, and built-in tests.

- **Data Interpretation:** The interpretation of data allows for conclusions of unacceptable or suspicious events that have happened (e.g., halt or failure condition), that are progressing (e.g., approaching a threshold of failure condition), or that can be expected to happen (i.e., in the absence of change, the failure condition will occur), including tracing to responsible entities to inform appropriate responses to events.

Caution must be taken with the use of design features that may hinder anomaly detection. Poorly designed lines of defense for defense in depth have been found to conceal emerging dangerous system states and conditions, especially from human observers [Saleh14]. The system design must minimize the difference between estimated system states and conditions and actual system states and conditions.

There are two approaches to anomaly detection:

- **Self-Anomaly Detection:** An entity has no dependency on another entity to detect an anomaly within the scope of its intended design. Self-anomaly detection usually involves an axiomatic or environmentally enforced assumption about its integrity. Typically, trusted elements have the capability for self-anomaly detection. This means that at the highest level of trustworthiness, an entity must be able to assess its internal state and functionality to a meaningful extent at various stages of execution. The detected anomalies must correlate to the trustworthiness assumptions placed on the entity.
- **Dependent Anomaly Detection:** An entity-of-interest is dependent on another entity for some or all anomalies that are detected. When an entity-of-interest relies on another entity for any portion of the assessment, that entity must be at least as trustworthy as the entity-of-interest.

REFERENCES: [Schroeder77]; [Smith12]; [Saleh14].

"System and component behaviors are monitored for anomalous operation. Adversaries innovate new attack methods to evade known-pattern detection screening. System and component behavior outside of normal expectations is a method-agnostic telltale."

-- **An Objective of the Security in the Future of Systems Engineering** [FUSE21]

E.2.2 Commensurate Protection

PRINCIPLE: *The strength and type of protection provided to a system element is commensurate with the most significant adverse effect that results from a failure of that element.*

Note: The strength and effectiveness of the protection for a system element must be proportional to the need. As the need increases, the protection of that element should also increase to the same degree. Need is derived from the most significant adverse effect associated with the system element or the trust that is placed in the element. The protection can come in the form of the system element's own self-protection, from protections provided by the system architecture, or from protection provided by other elements. The needed strength of protection is independent of these design choices (or others, such as distributed versus centralized design), a concept sometimes referred to as *secure distributed composition* [Neumann04]. Furthermore, confidence in the effectiveness of the protections provided to a system element should also increase commensurate to the need. This is addressed by the principle of *Commensurate Rigor*.

REFERENCES: [Neumann04]; [Levin07].

E.2.3 Commensurate Response

PRINCIPLE: *The system design matches the aggressiveness of an engineered response action's effect to the needed immediacy to control the effects of each loss scenario.*

Note: The selected response to a detected anomaly should consider three factors to determine the effect that the response has on the loss and the system:

- The expected effectiveness and aggressiveness of the response to directly address the anomaly and to prevent or limit the loss
- The direct, residual, or side-effect of the response on the system
- The opportunities that remain to take some other response action should the selected response fail to achieve the intended result

The response can be achieved by any combination of *fully manual*, *semi-automated*, *fully automated*, or *autonomous* means. However, the response action is distinct from the determination that a response is necessary and from the notification or signaling that invokes the response action.

A commensurate response requires consideration of the *response-effect-consequence* relationship associated with a specific loss. Ideally, for any given need for a response, a single action taken will be effective to resolve the loss concern and will have no associated adverse effect. Practically, due to complexity and the limits of certainty, the response action may not have the desired effect, may compound the problem, or may cause another problem. The balance required is one that determines if, when, and how a response action should be taken to be initially more aggressive or initially less aggressive. The severity of the problem and the time available for an effective response typically dictates a strategy for a continuum of responses, characterized by two extremes:

- **Graduated Response:** A graduated response is initially the least aggressive or impactful action possible to prevent the loss from continuing or escalating and does so with consideration of the possible side effects associated with the response action. The graduated response allows for taking increasingly more aggressive action should the loss situation persist or escalate.
- **Ungraduated Response:** An ungraduated response is the most aggressive and most impactful action possible to prevent the loss from continuing or escalating and does so without consideration of the possible side effects associated with the response action. The ungraduated response recognizes the severity of the loss as justifying the most aggressive action, even if that option provides no alternatives should it fail to have the intended or desired effect or if it causes other losses to occur.

Without early observability of possible loss, the option for a graduated response may not exist. Commensurate response is aided by early detection, which in turn increases the options for a graduated response.

REFERENCES: [Saleh14].

E.2.4 Continuous Protection

PRINCIPLE: *The protection provided for a system element must be effective and uninterrupted during the time that the protection is required.*

Note: The protection capability must be uninterrupted across all relevant system states, modes, and transitions for there to be assurance that the system can be effective in delivering the required capability while controlling loss. Continuous protection requires adherence to the following principles:

- **Trustworthy System Control:** Every controlled action is constrained by the mechanism, and the mechanism is able to protect itself from tampering. Sufficient assurance of the correctness and completeness of the mechanism can be ascertained from analysis and testing.
- **Protective Failure** and **Protective Recovery:** A protective state is preserved during error, fault, failure, and successful attack, as well as during the recovery of assets or of recovery to normal, degraded, or alternative operational modes.

Continuous protection applies to all configurations, states, and modes of the system, as well as the transitions between those configurations, states, and modes. The system design must ensure that protections are coordinated and composed in a non-conflicting and mutually supportive manner across the non-behavioral aspects of the system structure and the behavioral aspects of system function and data flow.

While the design for continuous protection applies for the entire time that the protection is required, there may be cases where, by design, protection capability is intentionally disabled (e.g., Battleshort[89] intentional override). The intentional disabling/override of protection is an exception case and, therefore, does not violate this principle. That is, the principle of *Continuous Protection* applies only for the entirety of time that the protection is required and not knowingly and intentionally disabled.[90]

REFERENCES: [Levin07].

E.2.5 Defense In Depth

PRINCIPLE: *Loss is prevented or minimized by employing multiple coordinated mechanisms.*

Note: The coordinated deployment of multiple protective mechanisms for a system helps to avoid single points of failure. The principle of defense in depth has several pillars:

- Multiple lines of defenses or barriers should be placed along loss scenario sequences.
- Loss control should not rely on a single defensive element.
- The successive barriers should be diverse in nature and include technical, operational, and organizational barriers.

Defense in depth requires the employment of coordinated mechanisms (active) within an architectural structure (passive) that achieves the *depth* characteristic.[91] Ideally, the initial lines of defense prevent loss, while subsequent lines of defense block loss scenario escalation and/or

[89] Battleshort is a switch used to bypass normal interlocks in mission-critical equipment (e.g., equipment that must not be shut down or the mission function will fail) during battle conditions [DOD 2007].

[90] However, the inclusion of a capability for intentionally disabling/overriding protection requires additional control features and devices and associated analysis for the enforcement of constraints to prevent the inadvertent actuation of the override capability.

[91] While the elaboration is limited to the machine, defense in depth may involve the combination of technical, operational, and organizational elements.

contain loss and potential consequences when needed. A defense-in-depth strategy examines loss scenarios for those points of opportunity to prevent or contain loss. It also leverages the opportunities to use active or passive mechanisms or constraints to meet loss control objectives.

The coordination of the multiple defense-in-depth mechanisms (i.e., combinations of structural, data, and control flow coordination) in conjunction with other design principles (e.g., *Anomaly Detection*, *Commensurate Response*) reflects a design strategy to satisfy the loss control objectives.

While defense in depth distributes the protection capability to many components, a defense-in-depth strategy may also consider a distributed composition to a line of defense. A protection capability provided by a single system component is a potential single point of failure or bottleneck to system performance. It may also raise other concerns. A distributed composition of a defense layer may provide additional options within the coordination of layers.

Defense in depth is, in part, a form of the principle of *Protective Failure*. It helps satisfy the objective that a failure of a system element should not result in an unacceptable loss. However, it does not satisfy the objective that a failure of a system element should not invoke another loss scenario.

REFERENCES: [Neumann04]; [Levin07]; [Jackson13]; [Saleh14].

E.2.6 Distributed Privilege

PRINCIPLE: *Multiple authorized entities act in a coordinated manner before an operation on the system is allowed to occur.*

Note: Distributed privilege[92] is a means to prevent a single authorized entity from performing an erroneous action, whether or not that action is performed with intent. Distributed privilege requires that an erroneous action can only be performed if multiple entities agree to do so, for either legitimate (e.g., override of the protection in extreme cases) or illegitimate purposes (e.g., collusion to intentionally take improper action). In the case of an attack on an operation, distributed privilege forces the adversary to target all of the entities to whom privilege is distributed.

Distributed privilege separates, divides, or in some other manner distributes the privileges required to perform an operation among multiple entities. The distribution of privilege includes a set of rules, conditions, and constraints that describe how multiple entities must interact through positive actions before a requested operation can proceed and be completed. The rules, conditions, and constraints may reflect combinations of the following, all of which require that multiple conditions be met for the operation to proceed:

- **Simultaneous Actions:** Multiple different authorized entities execute a command within a specified time window.
- **Sequenced Actions:** Multiple different entities interact within a linear sequence of actions where each successive action is enabled only by the successful completion of a prior action.
- **Parallel Actions:** Multiple entities execute sequences concurrently, and success is achieved either by a consensus of the results of each concurrent action or by voting among the participants.

[92] [Saltzer75] originally named this the *separation of privilege*. It is also equivalent to separation of duty.

REFERENCES: [Saltzer75]; [Levin07].

E.2.7 Diversity (Dynamicity)

PRINCIPLE: *The system design delivers the required capability through structural, behavioral, or data or control flow variation.*

Note: A system design that incorporates diversity helps to avoid common mode failures and introduces unpredictability to adversaries, thus complicating the planning and execution of where, when, and how to target their attacks. While the system behaviors that result from a design may be unpredictable from the viewpoint of the adversary, the design itself must be predictable and verifiable in achieving only the intended outcomes. The options for diversity include variety in the system structural and architectural design elements, the system functional and behavioral elements, the interfaces and interconnections between interfaces, the data and control flow, and the technology and component selection. Diversity can reside in:

- *Fixed or static characteristics of the system* (e.g., multiple instances of a system element, multiple communication channels)
- *Variable or dynamic characteristics of the system* (e.g., reconfiguration, relocation, refresh of system elements; random routing of data over different communication channels from source to destination; the ability to change aspects of the system behavior, structure, data, or configuration in a random but nonetheless verifiable manner)

Any design approach that includes diversity in structure, configuration, communications, protocols, and similar or dissimilar system elements (e.g., N-version, heterogeneity) increases uncertainty due to the increased complexity of the design and the behaviors and outcomes that stem from emergent effects, side-effects, and feature interaction. This drives the need for confidence that the design approach will deliver only the authorized and intended functional behavior, produce only the authorized and intended outcomes, and do so in a manner that allows for control over side-effects, emergence, and feature interaction.

Diversity options include intentionally designed regular or irregular changes in the system (e.g., implementing the concept of dynamicity).[93] This results in unpredictability and uncertainty to adversaries – complicating their attack planning – and can provide required performance despite other adversity. Dynamic change may refer to either shifting the target or shifting the behaviors of a target in performing its activities.

The uncertainty and diminished predictability associated with the employment of diversity and dynamicity in design can be problematic where it impedes or prevents having confidence that the system will function and produce outcomes only as authorized and intended. It is important to differentiate where the uncertainty lies: (1) uncertainty in how the system achieves an end objective (i.e., the means to an end) or (2) uncertainty that an objective will be achieved (i.e., achieving the end). A design that employs diversity and dynamicity must be based on acquiring confidence that the system will produce only the desired results despite uncertainty in knowing

[93] A design incorporating *dynamicity* can serve many purposes: (1) it complicates the attack planning of an adversary, (2) it reduces the potential for non-adversarial adversity to have an effect on the system, (3) it provides the capability and margin to deliver a required capability while reducing actual losses, and (4) it protects against the effects of an attack. An example of dynamicity is frequency hopping with wireless communications, which complicates the interception and jamming of signals.

exactly how the desired results are achieved. This constitutes a design trade that is specific to diversity- and dynamicity-based designs. Diversity may have a cost (e.g., hardware, software, maintenance, training, assurance) greater than the value or effectiveness that it provides.

REFERENCES: [Schroeder77]; [Jackson13]; [Moller08].

E.2.8 Domain Separation

PRINCIPLE: *Domains with distinctly different protection needs are physically or logically separated.*

Note: The separation of domains enables enhanced control and, therefore, protection of system function and the flow of data. Control relative to separated domains limits the extent to which an entity or domain is influenced by or is able to influence some other entity or domain, thereby enhancing the protection of a domain. This is achieved through the control of information flow and data between domains as well as control over the use of a system capability between domains.

The differing protection needs that are used to define domains may be thought of in terms of protecting the domain from influence by external entities (i.e., susceptibility) and protecting external entities from erroneous behavior that occurs within the domain (i.e., containment). This distinction may include separating critical functions from less critical functions, such as separating the flight control functions of a transport aircraft from the environmental control functions that maintain a safe environment for the cargo and passengers being transported.

Historically, domain separation has been used to enforce the separation of roles or privileges (i.e., least privilege). For example, a system may separate an "administrative" or "supervisor" domain from "user" domains. The administrative domain is accessible only by system administrators with appropriate privileges, and distinctly administrative functions may only be executed by administrators from the administrative domain. Similarly, data intended to only be accessed by administrators and administrative functions (e.g., system configurations) is stored and accessed only within that domain, ensuring needed protection of the data.

Domain separation requires a domain to be contained within its own protected subsystem so that elements of the domain are only directly accessible by procedures or functions of the protected subsystem. The concept of isolation enables the implementation of domain separation. Isolation limits the extent to which one domain can influence or can be influenced by other entities. The challenge is that the system elements within domains must at times interact with other elements and the environment to deliver a capability. Every interface that results from design decisions can diminish domain separation while achieving requirements for a system capability. External requests for resources or functions within protected subsystems are arbitrated at these interfaces. Firewall, data diodes, and cross-domain solutions (CDS) are examples of mechanisms that enable varying degrees of control over the interactions between separated domains.

Encryption is another mechanism often used to provide domain separation. For example, communication between distinct subsystems within a domain may be encrypted with a key that is known only to the subsystems within the domain. Where a common storage module or subsystem is used for multiple domains, encryption may be used to limit information access to the domain that owns the key to decrypt.

REFERENCES: [Smith12]; [Levin07].

E.2.9 Least Functionality

PRINCIPLE: *Each system element has the capability to accomplish its required functions but no more.*

Note: Susceptibility and vulnerability increase unnecessarily when a system element provides more functionality than is needed to achieve its intended purpose. Least functionality reduces the potential for susceptibility and vulnerability and also reduces the scope of analysis of the system element's trustworthiness and loss potential. The strictest interpretation of least functionality is to prohibit any system element functions that are not required. Where that is not possible or practical, the unnecessary functions of the system element should be disabled, disarmed, or put into a "safe" mode that prevents the functions from being used. In all other cases, mediated access can be used to prevent access to and use of the unneeded functions. An example of when it may not be possible or practical to avoid unnecessary functions is the use of commercial off-the-shelf (COTS) components. COTS components typically contain functions beyond those required to fulfill its intended purpose. In such cases, the components should be configured to enable only the functions that are required to fulfill its purpose and prohibit or restrict functions that are not required to fulfill its purpose.

REFERENCES: [Neumann04]; [Levin07].

E.2.10 Least Persistence

PRINCIPLE: *System elements and other resources are available, accessible, and able to fulfill their design intent only for the time for which they are needed.*

Note: Least persistence reduces susceptibility. It limits the extent to which functions, resources, data, and information remain present, accessible, and usable when not required, thereby reducing the opportunity for their inadvertent or unauthorized use, modification, or activation. The broadest interpretation of least persistence is to not install, instantiate, or apply power to system elements and resources until needed and to completely remove system elements or power from elements and resources when they are no longer required. Where that condition is not possible or practical, those system elements and resources should be fully disabled, disarmed, or put into safe mode to prevent their ability to function or to be used. At a minimum, *Mediated Access* should include constraints on the time and duration of their use.

Three conditions must be satisfied for an active system element or resource to be usable, with two of these conditions applying to non-active elements or resources:

- **Presence (active and non-active):** The system element or resource must be installed, loaded, residing in memory (software), and configured.
- **Accessible (active and non-active):** The system element or resource can be invoked, interacted with, or operated on.
- **Able to Function (active):** The system element or resource must be able to execute (i.e., powered on, enabled, or armed) to deliver a service or perform a function.

Least persistence is reflected in concepts such as sanitizing, erasing, clearing memory and storage locations; disabling, removing, and disconnecting network ports, system interfaces, and the services provided by system interfaces; powering off and unplugging hardware when not needed; and instantiating software just before needed and de-instantiating after it is no longer needed. Least persistence has added benefits that include simplifying the processes of:

- Cleansing the system element to remove corrupted aspects or side effects
- Re-establishing the system element to a known state (i.e., a refresh)
- Minimizing the period of time in which system elements are exposed to the environment, to attack, and to erroneous behavior

Where system elements or resources are removed and then restored as needed, there must be a trusted representation of the system element and a trusted ability to instantiate that system element within the time constraints for its use.

REFERENCES: [SP 800-160v2].

E.2.11 Least Privilege

PRINCIPLE: *Each system element is allocated privileges that are necessary to accomplish its specified functions but no more.*

Note: System elements can be implemented by entities such as hardware, firmware, software, and personnel. By design, the system must be able to limit the scope of a system element's actions. This has two desirable effects: (1) the impact of a failure, corruption, or misuse of the element is minimized, and (2) the analysis of the system element is simplified. A design driven by least privilege considerations results in a sufficiently fine granularity of privilege decomposition and the ability for the fine-grained allocation of privileges to human and machine elements. The application of the principle of least privilege means allocating the minimum (separate) privileges necessary to a system element according to the extent to which that element has a need to perform some function. This could include a need know, modify, delete, use, configure, authorize, start/enable, or stop/disable [Schroeder77]. In addition to its manifestations at the system interface, least privilege can also be used as a guide for the internal structure of the system itself, such as how to employ *Domain Separation*. One aspect of internal least privilege is to construct modules so that only the system elements encapsulated by the module are directly accessed or operated upon by the functions within the module. Elements external to a module that may be affected by the module's operation are indirectly accessed through interaction with the module that contains those elements.

REFERENCES: [Neumann04]; [Levin07]; [Saltzer75]; [Scroeder77].

E.2.12 Least Sharing[94]

PRINCIPLE: *System resources are shared among system elements only when necessary and among as few elements as possible.*

Note: Sharing via common mechanism and other means can increase the susceptibility of system resources (e.g., data, information, system variables, interfaces, functions, services) to unauthorized access, disclosure, use, or modification and can adversely affect the capabilities provided by the system. According to [Saltzer75], "Every shared mechanism (especially one involving shared variables) represents a potential information path between users and must be designed with great care to be sure it does not unintentionally compromise security." A design

[94] The historically well-known security design principle, *least common mechanism*, is an instance of least sharing. The principle of least common mechanism is described in [Popek74].

that employs least sharing helps to reduce the adverse consequences that can result from sharing system functions, state, resources, and variables among different system elements. A system element that corrupts a shared state or shared variables has the potential to corrupt other elements whose behavior is dependent on the state. Minimized sharing also helps to simplify the design and implementation [Lampson73].

There are two criteria that provide the basis for the application of the principle of least sharing: (1) share only if absolutely necessary, and (2) minimize sharing if allowed. The first criterion is a trade decision that factors in the cost and benefit of sharing resources against the increased exposure that results from the sharing. The second criterion is a constraint on the extent of sharing.

REFERENCES: [Popek74]; [Saltzer75]; [Lampson73]; [Neumann04] [Levin07].

E.2.13 Loss Margins

PRINCIPLE: *The system is designed to operate in a state space sufficiently distanced below the threshold at which loss occurs.*

Note: Margins refer to the difference between a conservative threshold at which the system is expected to operate while subjected to adversity and the point at which the adversity results in failure. Loss margins are created by engineered features put in place to maintain operational conditions and the associated adversity level at some distance (i.e., conservative threshold) from the estimated critical adversity threshold or loss-triggering threshold. Loss margins also allow for increased time to detect the need for a response action (see *Anomaly Detection*), to determine what the response action should be (see *Commensurate Response*), and to complete the selected response action. When there is uncertainty about the effectiveness of the response action, loss margins need to allow time to evaluate response effectiveness, determine any additional actions needed, and complete any selected actions.

Uncertainty may derive from the environment of operation, the design and realization of the system, the utilization and sustainment of the system, and the adversity presenting itself to the system. Loss margins are effective in addressing uncertainty about how and when a loss-triggering event occurs. Specifically, loss margins are effective in addressing uncertainty associated with:

- Intelligently designed and executed attacks, including attacks that persist and evolve over time
- Unknown, unquantified, and underappreciated susceptibilities, threats, hazards, vulnerabilities, and associated risks

For designs that incorporate loss margins, uncertainty about adversity makes determining the loss-triggering thresholds difficult. Loss margins for design should be determined with a balance between certainty (i.e., what has happened and can happen again) and uncertainty (i.e., what has not happened but can happen, or what has happened but can also happen in a different way). Loss scenarios that include loss escalation and an estimation of the critical threshold for loss occurrence are helpful in making design decisions that incorporate loss margins. Loss scenarios also help to determine the limits of adversity-driven decisions due to uncertainty in knowledge about the adversity (i.e., the adversity is insufficiently known or understood or is just unknown).

Sensitivity analyses must inform the determination of loss margins. Other factors for computing loss margins include system complexity, the use of newer technology or older technology in new ways, and the degree of new environments being introduced. An additional factor is the ability to complete comprehensive and effective testing. Limitations on system test coverage and effectiveness for all actual, simulated, or emulated adversity necessitate larger margins to account for the remaining uncertainty. The size of the margin may be reduced with time as unknown and underappreciated loss scenarios are uncovered and corrected, or the size may need to be increased over time as a malicious adversity capability matures in sophistication.

REFERENCES: [Saleh14]; [Moller08]; [NASA11]; [NASA14]; [Benjamin14]; [Pagani04].

E.2.14 Mediated Access

PRINCIPLE: *All access to and operations on system elements are mediated.*

Note: Mediated access is a foundational principle in the design of secure systems. The purpose of mediated access is to achieve the following:

- Place limits on access to and use of the system
- Reduce the possibility of loss escalation
- Reduce the extent to which loss escalates and propagates

Mediated access is based on the interaction between an entity and a target system element and has two aspects:

- **Access to the System Element:** The requesting entity only has authorized access to a target system element.
- **Use of the System Element:** The requesting entity is only allowed to perform authorized operations on the target system element.

Mediated access has two parts: (1) a policy-based access mediation decision and (2) the enforcement of the access mediation decision. The access mediation decision may include conditional constraints that further restrict access (e.g., role, time of day, system state or mode, or duration of operation). If access is not sufficiently mediated, there is no possibility of limiting how system elements (including human and machine elements) interact to ensure that only authorized behaviors and intended outcomes result.

Mediated access is achieved by an access mediation control mechanism. Seminal computer security work defined the *reference validation mechanism* as the generalized form of any mechanism that is an implementation of the reference monitor concept (Section D.4.2). The reference monitor provides the design assurance basis for demonstrating the trustworthiness of a mediated access control mechanism. The essential design criteria (Section D.4.2) provide a refinement to extend the generalized reference monitor concept. Mediated access may enforce the constraints described in the principles of *Distributed Privilege*, *Least Privilege*, and *Least Sharing*.

Efficiently mediated access refers to using a *least common mechanism* for mediating access. Mediating access is often the predominant security function within a secure system and may result in performance bottle necks if not designed and implemented correctly. The use of least common mechanism is one means to help reduce bottle necks [Levin07].

REFERENCES: [Saltzer75]; [Neumann04]; [Levin07]; [Neumann17]; [Anderson72]; [Saleh14].

E.2.15 Minimize Detectability

PRINCIPLE: *The design of the system minimizes the detectability of the system as much as practicable.*

Note: A system that is not discoverable, observable, or trackable by an adversarial threat or exposed to such a threat is less prone to a targeted attack. Minimizing detectability drives engineering design decisions to eliminate or reduce exposures such as unnecessary interfaces, access points, footprints, and emanations, thereby reducing susceptibility to adversarial threat actions. Interfaces and access points have the effect of exposing the system to intentional adversity (i.e., attacks) and non-intentional adversity (i.e., faults, errors, incidents, accidents). Yet interfaces and access points are necessary to compose system elements to deliver required capabilities, and some duplication of interfaces and access points is needed to avoid single points of failure. System design must balance the need for interfaces with the susceptibility that results from the interface being exposed, discovered, and observed. Every interface, whether internal or external, constitutes an exposure that must be considered.

Minimizing detectability reduces the ability of an adversary to observe and discover information about the system to craft and execute attacks. This includes detection of a system's location, presence, and movement (e.g., due to emissions, signatures, or footprints). There are various ways that a system may be detectable, including heat emission, electronic magnetic (EM) emissions, sound, vibrations, reflecting radar waves or light, or the response to stimulus (e.g., a response to an Internet Control Message Protocol [ICMP] echo request or "ping"). There are specific forms or means to minimize detectability, including camouflage, stealth, low probability of intercept/low probability of detect (LPI/LPD) waveforms (for radios), and frequency hopping.

REFERENCES: [Bryant20]; [Ball03]; [SP 800-160v2].

E.2.16 Protective Defaults

PRINCIPLE: *The default configuration of the system provides maximum protection effectiveness.*

Note: The configuration of the system includes the parameters for system functions, data, interfaces, and resources that determine how the system behaves and the outcomes it produces. Protective defaults guarantee that the "as shipped" system configuration and parameters prioritize the achievement of loss control objectives over the ability to deliver a required system capability and performance without dependence on human intervention. Protective defaults require conscientious action to establish the system configuration and parameters that deliver the required capability and performance in a manner that provides *Commensurate Protection* against loss. Protective default configurations for systems include constituent subsystems, components, and mechanisms. The principles of Protective Failure, Protective Recovery, and Continuous Protection parallel this principle to provide the ability to detect and recover from failure.

REFERENCES: [Saltzer75]; [Neumann04]; [Levin07].

E.2.17 Protective Failure

PRINCIPLE: *A failure of a system element neither results in an unacceptable loss nor invokes another loss scenario.*

Note: Protective failure is the aspect of continuous protection that ensures that a protection capability is not interrupted during a failure and that the effect of the failure is constrained. Two aspects of protective failure must be satisfied to achieve the intended effect:

- **Avoid Single Points of Failure:** The failure of a single system element should not lead to unacceptable loss. Unacceptable loss should only occur in the case of multiple independent malfunctions – a safety principle known as *single failure criterion*. The principle of *Defense in Depth* can help achieve this aspect of protective failure.
- **Avoid Propagation of New Failure:** If unmitigated, failures in the system can result in propagating, cascading, or rippling effects on the system. These effects can be addressed if the remaining protections remain effective to prevent the originating failure from causing additional failures. The principle of *Defense in Depth* does not address the propagation of failure by invoking a new loss scenario and, therefore, does not help achieve this aspect of protective failure without additional analysis.

Protective failure applies to discrete system elements, aggregates of system elements, and the systems abstraction. Protective failure seeks to limit the effect of a failure to the extent practicable and, in doing so, minimize the introduction of new loss possibilities. Protective failure is able to limit the extent to which a failure is able to advance loss scenarios associated with the failure, including cascading losses; trigger a different loss scenario; or create a new loss scenario. Efforts to avoid or limit failures may themselves degrade system performance, a form of failure. Thus, system designers may need to consider trade spaces between possible adverse effects and system performance.

REFERENCES: [Neumann04]; [Jackson13]; [Saleh14]; [Moller08]; [Levin07].

E.2.18 Protective Recovery

PRINCIPLE: *The recovery of a system element does not result in nor lead to unacceptable loss.*

Note: Protective recovery is an aspect of *Continuous Protection* that ensures that a protection capability is not interrupted during recovery from actual or impending failure. Protective recovery is applied to discrete system elements, aggregates of system elements, and the system. To the extent practicable, any recovery from impending or actual failure to resume normal, degraded, contingency or alternative operation, or the recovery of other asset losses should not (1) advance the loss scenario that is the target of the recovery, (2) trigger other loss scenarios, or (3) create new loss scenarios. The practicable aspect of this principle recognizes that for some recovery efforts to be successful, they may degrade system performance, which is a form of loss. Protective recovery is an aspect of the response strategy for the system. Thus, graduated and ungraduated considerations of *Commensurate Response* apply to best suit expediency in the need for a protective recovery.

REFERENCES: [Schroeder77]; [Neumann04]; [NASA11]; [Levin07].

E.2.19 Redundancy

PRINCIPLE: *The system design delivers the required capability by the replication of system functions or elements.*

Note: Redundancy employs multiples of the same system elements, data and control flows, or paths to avoid single points of failure. Redundancy requires a strategy for how multiple system elements are used individually or in combination (e.g., load-balancing, fail-over, concurrently, backup, voting, agreement, consensus). Redundant solutions are susceptible to common mode failure (i.e., a single event that results in the same or equivalent elements failing in the same manner). The cause of the failure may occur with or without intent. *Diversity* is a means to address the concerns of common mode failure.

REFERENCES: [Schroeder77]; [Neumann04]; [Jackson13]; [Moller08].

APPLICATION OF DESIGN PRINCIPLES TO COMMERCIAL PRODUCTS

For commercial products to be trustworthy commensurate with their criticality, security design principles should be selected and applied appropriately throughout the products' system life cycle. Each design principle must be assessed for its relevance, applicability, and validity. The security design principles described in this appendix have been demonstrated by industry in past work and have previously been codified into national and international standards and guidance documents, including the Department of Defense *Trusted Computer System Evaluation Criteria (TCSEC)* and ISO/IEC 15408, *Common Criteria for Information Technology Security Evaluation*.

Many commercial products have been designed, developed and evaluated against specifications from those standards and guidelines up to and including the highest levels of assurance (e.g., TCSEC Class A1 and Class B3). These products represent use cases of trustworthy components and systems that have been verified to be highly resistant to penetration from determined adversaries and, in the case of TCSEC Class A1, distinguished by substantially dealing with the problem of subversion of security mechanisms. To merit the trust of consumers, commercial products must demonstrate – in a manner that can be independently verified – that the security design principles articulated in this appendix have been applied to produce components and systems that are both sound and logically coherent with respect to security.

APPENDIX F

TRUSTWORTHINESS AND ASSURANCE

REDUCING UNCERTAINTY AND BUILDING CONFIDENCE IN THE SYSTEM

The determination that a system[95] is trustworthy is based on the concept of *assurance*. Assurance is the grounds for *justified confidence* that a claim or set of claims has been or will be achieved [ISO 15026-1]. Justified confidence is derived from objective evidence that reduces uncertainty to an acceptable level and in doing so, reduces risk.[96] Evidence is acquired through the application of rigorous engineering verification methods.[97] The evidence must be relevant, accurate, credible, and of sufficient quantity to enable reasoned conclusions and consensus among subject-matter experts that the claims are satisfied. The relationship between evidence and claims can be represented in various ways. These approaches are discussed in Section F.2.

> *"The trust we place in our digital infrastructure should be proportional to how trustworthy and transparent that infrastructure is and to the consequences we will incur if that trust is misplaced."*
>
> -- ***Executive Order (EO) on Improving the Nation's Cybersecurity*** [EO 14028]
> May 2021

F.1 TRUST AND TRUSTWORTHINESS

The concepts of *trust* and *trustworthiness* are foundational to trustworthy secure design, the decisions made to grant trust, and the extent to which trust is granted based on *demonstrated* trustworthiness. Trust is a belief that an entity meets certain expectations, and therefore, can be relied upon. The terms *belief* and *can* imply that trust may be granted to an entity whether the entity is trustworthy or not. A trustworthy entity is one for which sufficient evidence exists to support its claimed trustworthiness. Thus, trustworthiness is the demonstrated ability and, therefore, worthiness of an entity to be trusted to satisfy expectations. Trustworthiness, being something demonstrated, is based on evidence that supports a claim or judgment of an entity being worthy to be trusted [Schroeder77] [Neumann04] [Levin07].

Trust in an entity can occur without a basis for or knowledge of the entity's trustworthiness. Trust may occur because: (1) there is no alternative (e.g., an individual trusts the components involved in an Internet transaction without knowing anything about the components), (2) the need for trustworthiness is not realized and occurs de facto, or (3) other reasons [Neumann17].

[95] As discussed in Chapter Two, a *system of interest* can be a system, sub-system, component, system of systems, network, as well as an infrastructure.

[96] Section F.2 describes the relationship between uncertainty and risk.

[97] Verification methods include demonstration, inspection, analysis, and testing. These verification methods support decision-making throughout the system life cycle, including decisions for major reviews and for system acceptance, approval, or authorization. Additionally, there are other types of validation activities, such as the validation of requirements prior to their incorporation into a configuration-controlled requirements baseline.

Since trust is not necessarily based on a judgment of trustworthiness, the decision to trust an entity should consider the consequences, effects, and impacts of trust *expectations* not being fulfilled because of non-performance, whether due to failure, deficiency, or incompetence. Ideally, the criteria to grant trust is used to determine the trustworthiness of an entity. Trust that is granted without establishing the required trustworthiness is a significant contributor to risk.

F.1.1 Roles of Requirements in Trustworthiness

Trustworthiness judgments are based on criteria that express the need to trust. This need must be transformed into requirements in the same way that capability, performance, security, and other needs are transformed into requirements. The trustworthiness judgments are meaningful only to the extent that the trustworthiness-relevant requirements accurately reflect the problem, accurately define the solution, and can be verified as being satisfied by the solution. Trustworthiness requirements about security derive from the protection needs, priorities, constraints, and concerns associated with the ability of the system to achieve authorized and intended behaviors and outcomes, deal with adversity, and control loss. The requirements also address the measures used to assess trustworthiness and the evidentiary data required to substantiate conclusions about trustworthiness and granting trust based on trustworthiness. The discipline of *requirements engineering* provides the methods, processes, techniques, and tools for this to occur.

> *"A meaningful claim of trustworthiness cannot be based on an isolated demonstration that the system contains protection capability assumed to be effective or sufficient. Instead, conclusions about protection capability must have their basis on evidence that the system was properly specified, designed, and implemented with the rigor needed to deliver system-level function, in a manner deemed to be trustworthy and secure*." [Neumann04]

F.1.2 Design Considerations

The design for a trustworthy secure system requires the rigorous application of principled engineering concepts and methods supported by evidence that provides assurance that all security-related claims about the system are satisfied (Section F.2).[98] There are several considerations that apply to achieving trustworthiness in system design:

- **Composition**

  Trustworthiness judgments themselves are compositional. They must align with how the set of composed elements provides a system capability. The way the system is composed from its system elements must include the application of the design principle of *Compositional Trustworthiness* coupled with the principle of *Structured Decomposition and Composition* to the extent practical.

[98] Constraints and claims are expressed in terms of functional correctness, strength of function, concerns for asset loss and consequences, and the protection capability derived from adherence to standards or from the use of specific processes, procedures, or methods.

- **States, Modes, and Transitions**

  Ideally, the implemented system design will result in a system that continually remains in secure states and modes, with secure transitions between states and modes. Realistically, the system will have insecure and indeterminant (unknown if secure or insecure) systems states and modes. The design must account for these cases and provide the capability to transition from insecure states and modes to secure states and modes (see *Protective Recovery*). In short, the system design must account for behaviors and outcomes that comprise secure, insecure, and indeterminant states, modes, and transitions.

- **Failure Propagation**

  All systems fail. When a failure occurs, it should not trigger or invoke some other failure scenario or create a new failure scenario (see *Protective Failure*). Design without single points of failures (see *Redundancy*), including not having common mode failures (see *Diversity*), can isolate system element failures while providing required system capabilities. Additionally, the response to failure should not lead to loss or other failures (see *Protective Recovery*).

- **Anomaly Detection**

  Anomaly Detection provides situational awareness that allows the system to make decisions and provide recommendations for corrective action to account for actual and potential deviations from the accepted norms.

- **Trades**

  Not every system element may have trustworthiness that is sufficient for its intended purpose. A deficiency in trustworthiness can result from:

  - Technical feasibility and practicality issues
  - Cost and schedule issues of what is feasible and practical
  - Limits of certainty (i.e., what is not known, what cannot be known, and what is underappreciated [known or could be known but dismissed prematurely])

  The *trade space* is the application of the combined set of trustworthiness and loss control principles that provides a basis for making the necessary design decisions to maximize the trustworthiness of individual system elements and the trustworthiness of aggregates of elements that must be trusted. For example, in addressing the feasibility and practicality of cost and schedule issues described above, the design principle of minimizing the number of system elements that must be trusted (see *Minimized Trusted Elements*) is applied. This reduces the size and scope of the effort, and potentially reduces the expense to generate evidence of trustworthiness.

F.2 ASSURANCE

Assurance is the grounds for justified confidence that a claim or set of claims has been or will be achieved [ISO 15026-1]. Assurance is a complex and multi-dimensional property of the system that builds over time. Assurance must be planned, established, and maintained in alignment with the system throughout the system life cycle.

Judgments of adequate security should be based on the level of confidence in the ability of the system to protect itself against asset loss and the associated consequences across all forms of adversity.[99] It cannot be based solely on individual efforts, such as the demonstration of compliance, functional testing, or adversarial penetration tests. Judgments include what the system cannot do, will not do, or cannot be forced to do. These judgments of non-behavior must be grounded in sufficient confidence in the system's ability to correctly deliver its intended function in the presence and absence of adversity and to do so when used in accordance with its design intent.

The needed evidentiary basis for such judgments derives from well-formed and comprehensive evidence-producing activities that address the requirements, design, properties, capabilities, vulnerabilities, and effectiveness of security functions. Testing is one of several verification activities. The evidence acquired from these activities informs reasoning by qualified subject-matter experts to interpret the evidence to substantiate the assurance claims made while considering other emergent properties that the system may possess.

VENEER SECURITY

Veneer security is security functionality provided without corresponding assurance so that the functionality only ***appears*** to protect resources when, in fact, it does not. Veneer security results in a false sense of security and, in fact, increases risk due to the uncertainty about the behavior and outcomes produced by the security functionality in the presence and absence of adversity. Veneer security must be avoided [Saydjari18].

Compliance is a form of "veneer security." While compliance may have an important *informing* role in judgments of trustworthiness, compliance-based judgments – like other forms of veneer security – do not suffice as the sole evidentiary basis for assurance and the associated judgments of trustworthiness.

F.2.1 Security Assurance Claims

From a security perspective, a top-level claim addresses freedom from the conditions that cause asset loss and the associated consequences by ensuring the system achieves only authorized and intended system behaviors and outcomes. Supporting claims include the completeness and accuracy of stakeholder and system requirements, a sound approach to design, the proper implementation of the design, and the proper use and maintenance of the system.

When applied to security, the top-level claim is that the *system* will adequately contribute to freedom from the conditions that cause asset loss and the associated consequences. The top-level security claim decomposes into claims about the design, implementation, requirements, methods, and adversities in a structured manner that demonstrates that the design adequately contributes to ensuring only authorized and intended system behaviors and outcomes.

[99] The term adversity refers to those conditions that can cause a loss of assets (e.g., threats, attacks, vulnerabilities, hazards, disruptions, and exposures).

Security assurance claims reflect the desired attributes of a trustworthy secure system. These claims are derived from concerns about the completeness and accuracy of stakeholder and system requirements,[100] enforcement of the security policy, proper implementation of the design, proper maintenance of the system, the usability of the system,[101] and the avoidance, minimization, and mitigation of defects, errors, and vulnerabilities.[102] There may also be other claims involving the ability to exhibit predictable behavior while operating in secure states in the presence and absence of adversity and the ability to recover from an insecure state. Claims can be expressed in terms of functional correctness, strength of function, and the protection capability derived from the adherence to standards and/or from the use of specific processes, procedures, and methods.

LEARNING FROM SAFETY

The NASA System Safety Handbook [NASA11] describes the relevant *claims* to be met in terms of the top-level claim that the system is adequately safe with *subclaims*, including the system is designed to be as safe as reasonably practicable, built to be as safe as reasonably practicable, and operated as safely as reasonably practicable.

F.2.2 Approaches to Assurance

There are three general approaches to assurance. These approaches vary based on type of evidence, how the evidence is acquired, the strength of the judgments made based on the evidence, and the extent to which the assurance matches decision-making needs. From weakest to strongest, the assurance approaches are *axiomatic*, *analytic*, and *synthetic*.

- **Axiomatic Assurance** (assurance by assertion) is based on beliefs accepted on faith in an artifact or process. The beliefs are often accepted because they are not contradicted by experiment or demonstration. Axiomatic assurance is not suited to complex scenarios.
    - Demonstration of conformance and compliance are types of axiomatic assurance. While useful, they are not well-suited as the sole basis of assurance for complex scenarios.

[100] Claims are not expressed solely as a restatement of the security functional and performance requirements. Doing so only provides assurance that the security requirements are satisfied with the implicit assumption that the requirements are correct, provide adequate coverage, and accurately reflect stakeholder needs and concerns.

[101] [Anderson20] observes that most system failures have a human component, and that assurance must consider human frailty. Furthermore, [Leveson11] notes that operator behavior is a product of the environment (including its systems) in which it occurs.

[102] Not all vulnerabilities can be mitigated to an acceptable level. There are three classes of vulnerabilities in systems: (1) vulnerabilities whose existence is known and either eliminated or made to be inconsequential, (2) vulnerabilities whose existence is known but that are not sufficiently mitigated, and (3) unknown vulnerabilities that constitute an element of uncertainty. That is, the fact that the vulnerability has not been identified should not give increased confidence that the vulnerability does not exist. Determining the effect of vulnerabilities that are in the delivered system and the risk posed by those vulnerabilities and accepting that there is uncertainty about the existence of a vulnerability that will only become known over time are important aspects that are addressed by assurance.

- **Analytic Assurance** (assurance by test and analysis) derives from testing or reasoning to justify conclusions about properties of interest. Belief is relocated from an artifact or process to trust in some method of analysis. The feasibility of establishing an analytic basis depends on the amount of work involved in performing the analysis and on the soundness of any assumptions underlying that analysis. Analytic methods are most relevant in a model that spans *all* relevant uses and *all* interfaces to the environment. That is, the model must not ignore too many details.
    - Testing demonstrates the presence but not the absence of errors and vulnerabilities. Testing and analyses will have *uncertainty* that cannot be ignored, especially when they lack comprehensiveness. Uncertainty contributes to risk.
- **Synthetic Assurance** (assurance by structured reasoning) derives from the method of composition of the "components of assurance" (i.e., the assurance derives from the manner of *synthesis* of the constituent parts). It requires that assurance be a consideration at every step of design and implementation, from the smallest components to the final subsystem realization.
    - The assurance case described in [ISO 15026-2] is an example of structured reasoning (also see Section 2.5.3). Structured reasoning serves to fill the gaps associated with the axiomatic and analytic assurance approaches. Since synthetic assurance is based on expert judgment of available evidence, it is not complete. However, synthetic assurance does further reduce uncertainty and thus reduces risk.

ASSURANCE CASE

An *assurance case* is a reasoned, auditable artifact that is created to support the contention that a top-level claim is satisfied. The assurance case includes systematic argumentation, evidence, and explicit assumptions that support the claim.

An assurance case contains the following elements [ISO 15026-2]:

- One or more claims about properties
- Arguments that logically link the evidence and any assumptions
- A body of evidence
- Justification of the choice of a top-level claim and the method of reasoning

[NASA17] found that assurance cases have numerous advantages over other means for obtaining confidence, such as in the areas of comprehension, informing needed allocation responsibilities, information organization, and robust due diligence. These advantages were larger in areas with otherwise insufficient methods to achieve high assurance. Additionally, assurance cases were determined to be more efficient for complex and novel systems, as well as systems in need of high assurance.

Many formalizations and tools for building assurance cases have been developed in recent years, including the Goal Structuring Notation (GSN) [GSNCS18] and NASA's AdvoCATE: Assurance Case Automation Toolset [NASA19].

Assurance in the system depends on the *quality* of the evidence used in arguments that demonstrate that claims about the system are satisfied. Assurance evidence can be obtained directly through measurement, testing, observation, or inspection. It can also be obtained indirectly through analysis, including the analysis of data obtained from measurement, testing, observation, or inspection. Evidence must have sufficient quality in accuracy, credibility, relevance, rigor, and quantity. The accuracy, credibility, and relevance of evidence should be confirmed prior to its use. For example, some evidence can support arguments for strength of function, others for negative requirements (i.e., what will not happen), and still other evidence for qualitative properties.

F.2.3 Assurance Needs

Assurance is a need that is engineered and satisfied similar to the need to engineer the system capability to satisfy capability needs. Assurance needs for trustworthy secure systems are grounded in the concerns of loss and adverse effects due to intentional and unintentional adversity (see the design principles of *Commensurate Rigor*, *Commensurate Trustworthiness*, and *Substantiated Trustworthiness*). Assurance needs include the evidence-basis for reasoning, the degree of rigor to acquire and interpret the evidence, and the selection of the methods, tools, and processes used throughout the system life cycle. Like capability and performance needs, assurance needs, expectations, priorities, and constraints should be expressed as system requirements and achieved, tracked, and maintained within *systems engineering* as such.

CONFIDENCE MAY BE NEGATIVE

Confidence that is obtained through analysis is not necessarily positive. Assurance evidence can support a compelling argument that counters a stated claim, as well as a conclusion that there is insufficient confidence to support a trustworthiness decision. That is, the system or some part of the system is not sufficiently trustworthy and should *not* be trusted relative to its specified function without further action to establish a sufficiently credible and reasoned evidence base for its use. Alternatively, a risk analysis and risk treatment may be performed [ISO 16085].

Assurance needs determine the type of evidence and the rigor associated with the activities, methods, and tools used to acquire the evidence to satisfy the following cases:

- **What is done:** The realization of the design for a secure system
- **The means to accomplish what is done:** The methods, processes, and tools employed (driven by rigor and assurance objectives) to realize the design for a secure system
- **The results of what is done:** The substantiated effectiveness of the realized design of the secure system

Assurance needs can vary and constitute a *trade space* that must be managed similar to how capability and performance needs can vary. The degree of rigor is the primary means of varying assurance. As shown in Figure F-1, a direct relationship exists between the degree of rigor and assurance and the stakeholder's assessment of the effects of asset loss. The assurance trade space includes the following considerations:

- Cost, schedule, and performance
- Architecture and design decisions
- Selection of technology and solutions
- Selection and employment of methods and tools
- Qualifications necessary for subject-matter experts

Requirements analysis across stakeholder and system requirements determines the *threshold* degree of rigor that is required. When a system cannot practicably meet the needed degrees of rigor, stakeholders should have a means to determine if they will accept the associated risk.

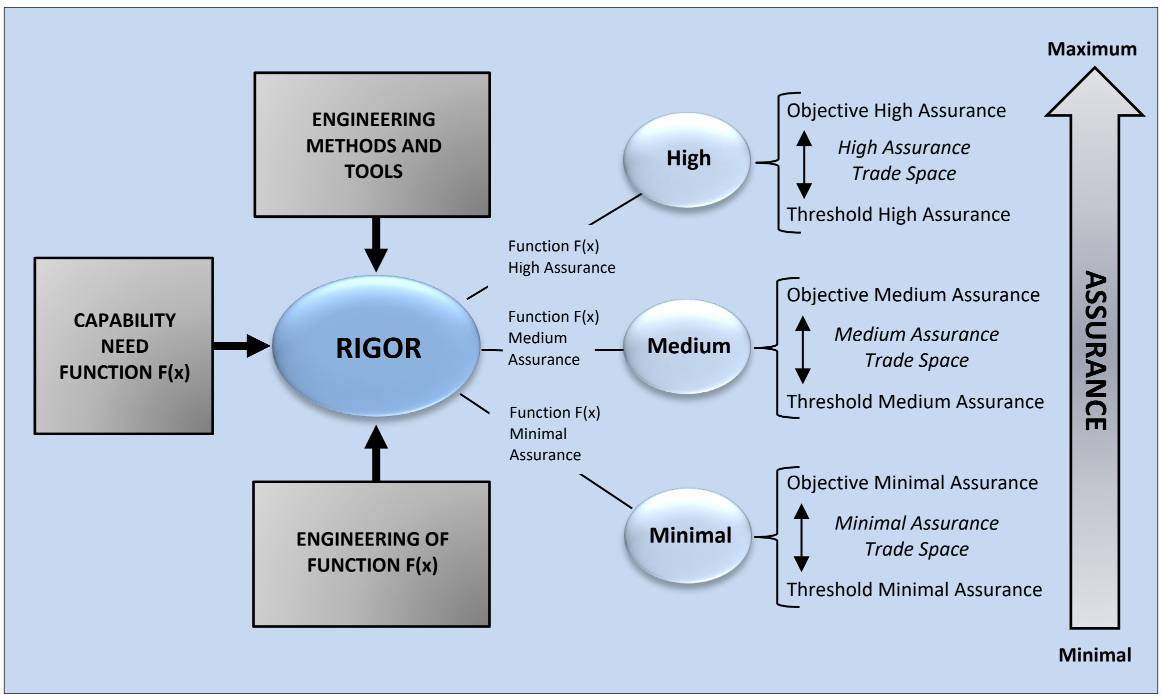


FIGURE F-1: ASSURANCE AND DEGREE OF RIGOR IN REALIZING A CAPABILITY NEED

The highest levels of rigor across systems often requires formal methods—techniques that model systems as mathematical entities to enable rigorous verification of the system's properties through mathematical proofs. Formal methods depend on formal specifications (i.e., statements in a language whose vocabulary, syntax, and semantics are formally defined) and a variety of models including a formal security policy model (i.e., a mathematically rigorous specification of a system's security policy [Appendix C]).

Due to the cost and complexity associated with formal methods, such methods are typically limited to engineering efforts where only the highest levels of assurance are needed, such as the formal modeling, specification, and verification of security policy and the implementation that enforces the policy (Section D.4.2). In this case, the security policy model is verified as complete for its scope of control and as self-consistent. The verified security policy model then serves as a foundation to verify the models of the design and implementation of the mechanisms providing for decision-making and the enforcement of those decisions.

DOES DEFENSE IN DEPTH INCREASE TRUSTWORTHINESS?

[Levin07] noted:

"The notion of defense in depth describes security derived from the application of multiple mechanisms (e.g., to create a series of barriers against an attack by an adversary). However, there is no theoretical basis to assume that defense in depth, in and of itself, could imply a level of trustworthiness greater than that of the individual security components. Without a sound security architecture and supporting theory, the nonconstructive nature of these approaches renders them equivalent to temporary patches."

Moreover, [Saleh14] notes that poorly designed *defense in depth* layering can actually conceal emerging dangerous system states and conditions. For more information on the proper use of the principle for trustworthy secure design, *Defense In Depth*, see Appendix E.